

1/40

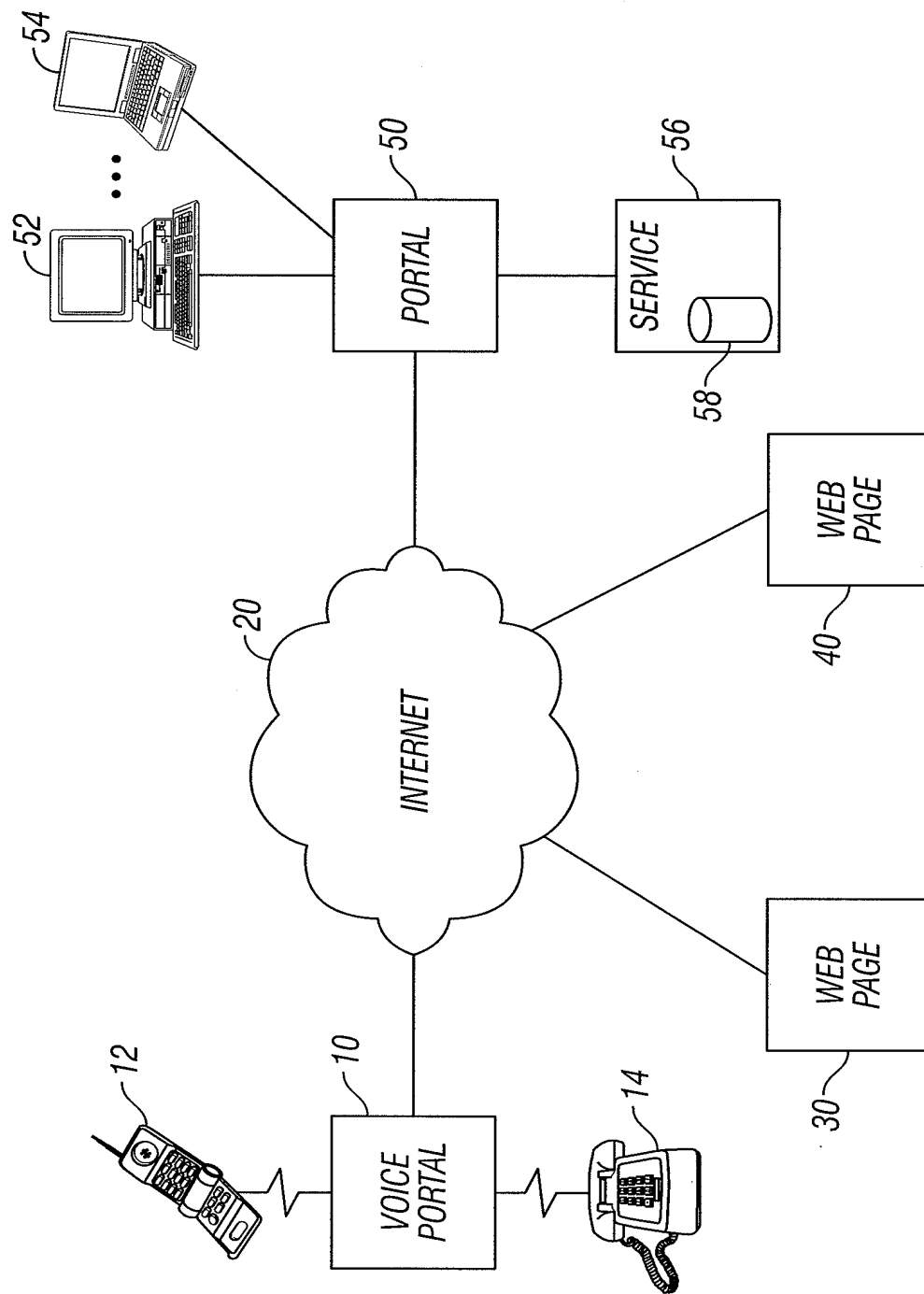


FIG. 1

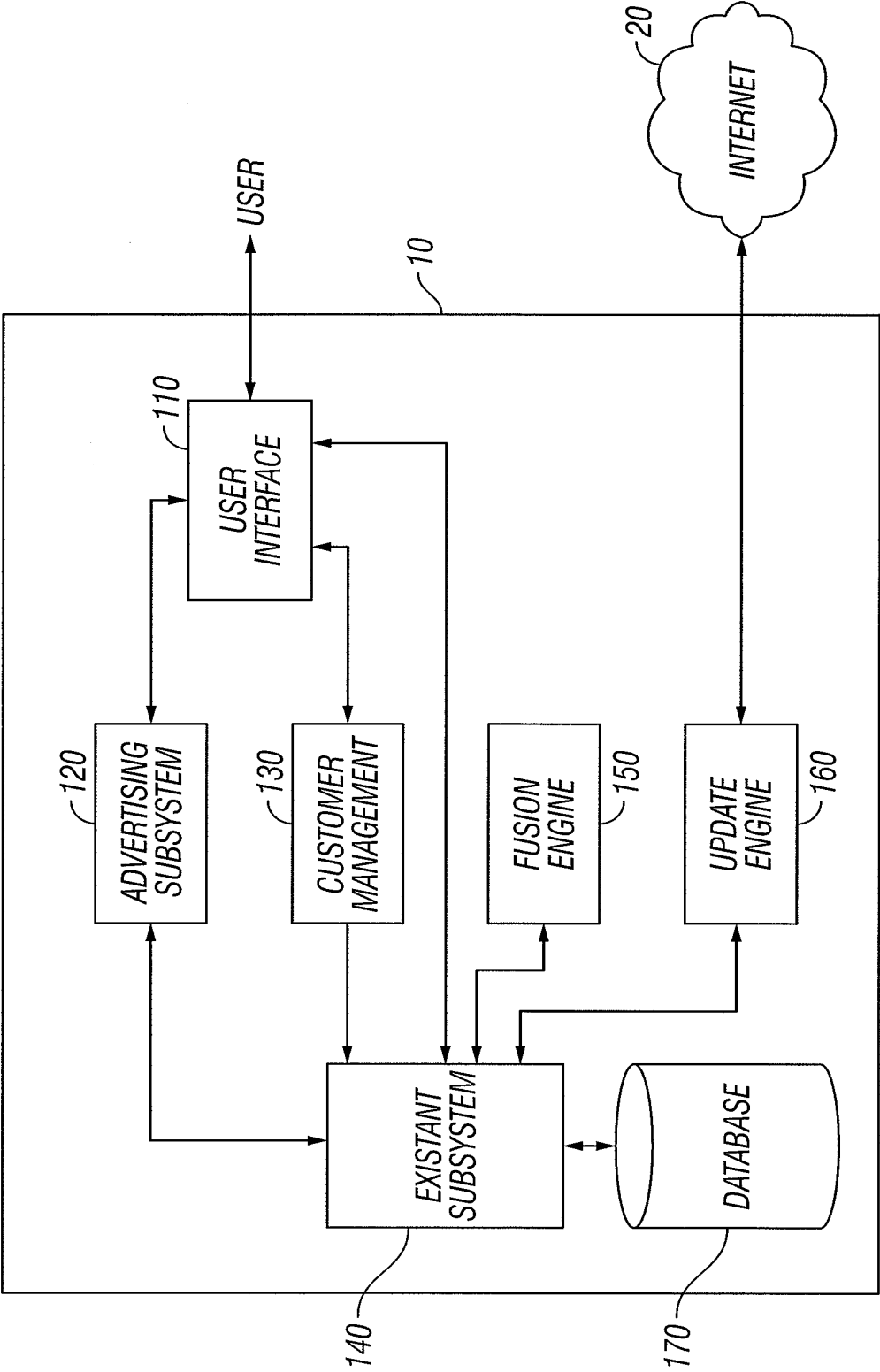


FIG. 2

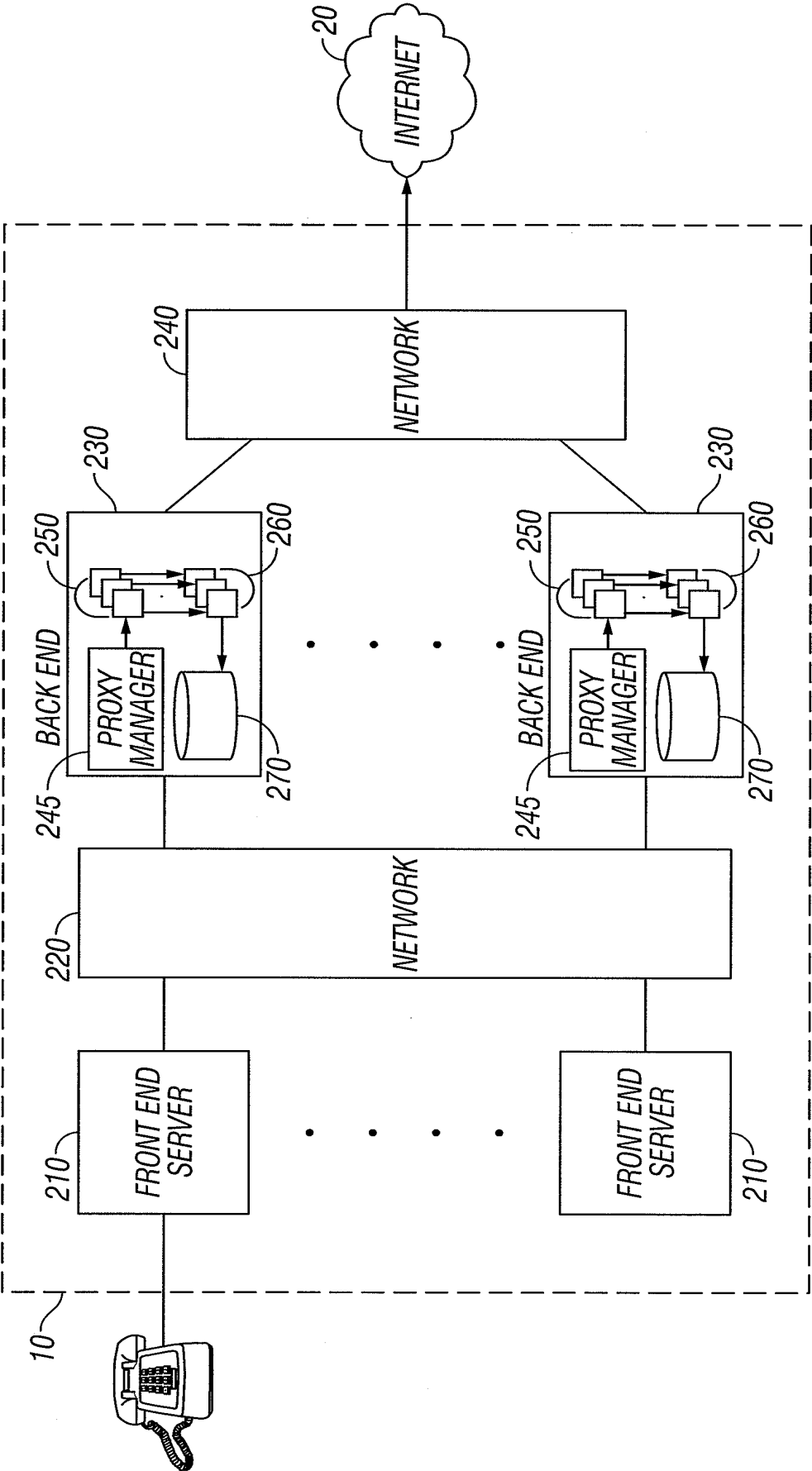


FIG. 3

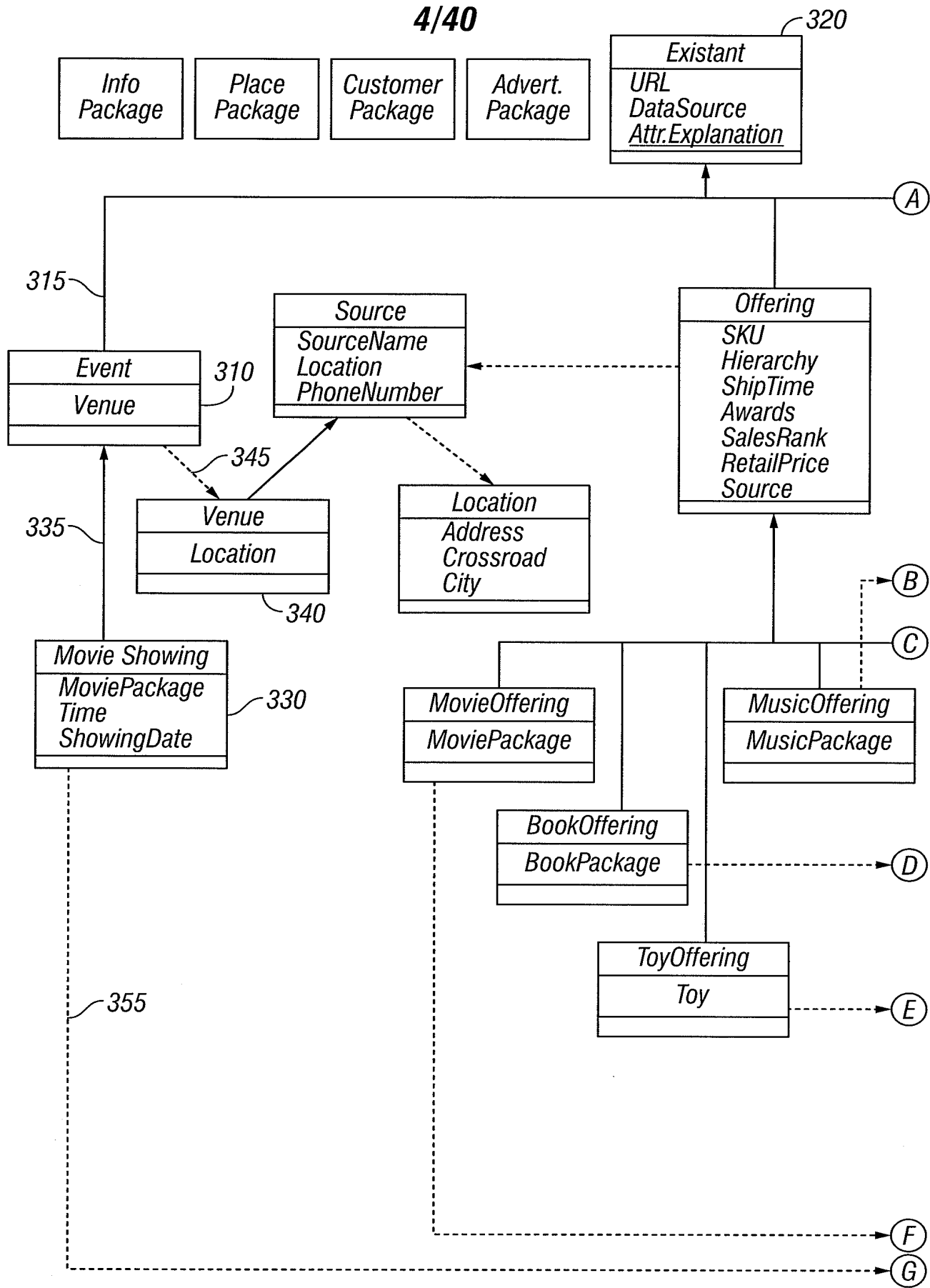
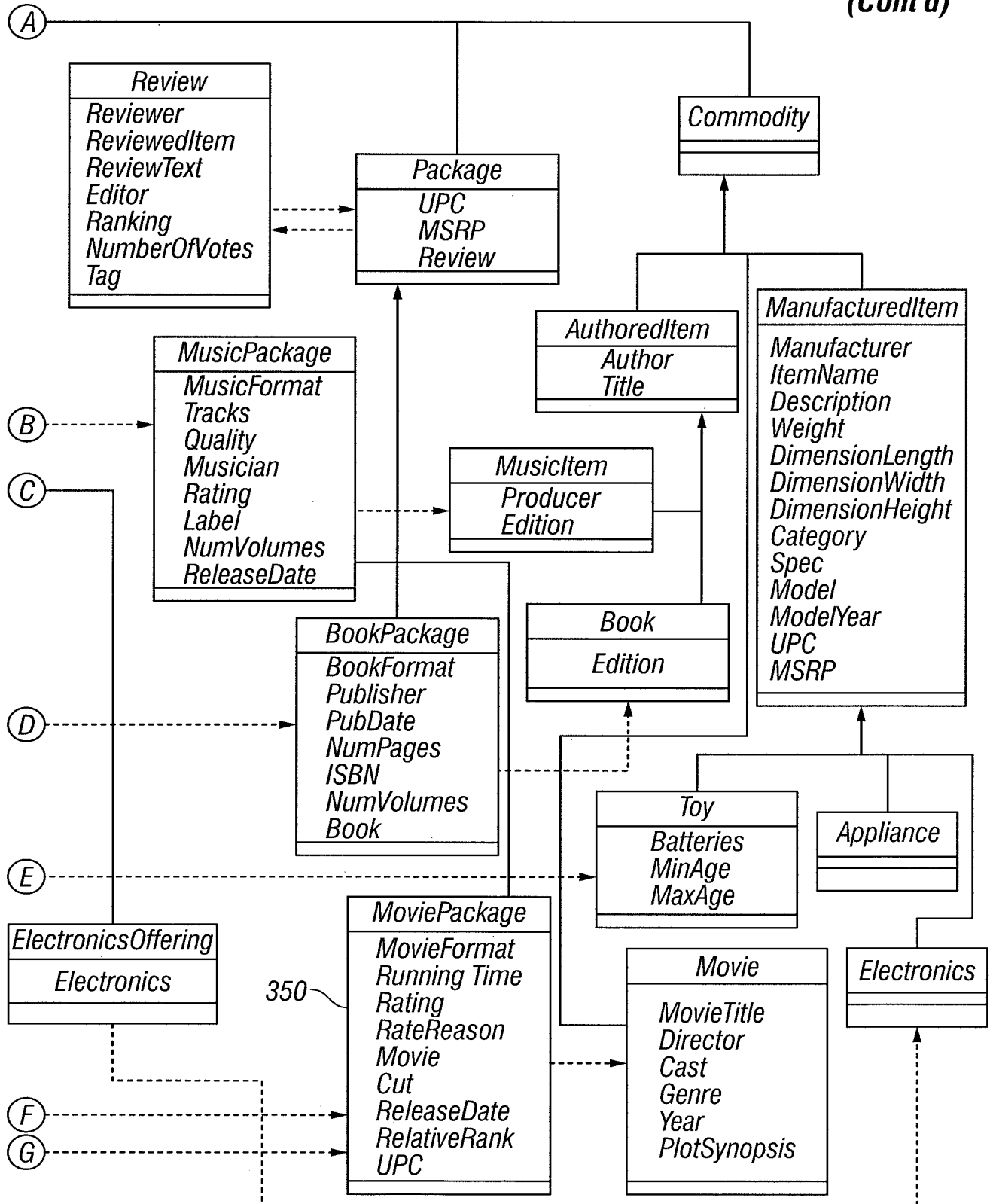
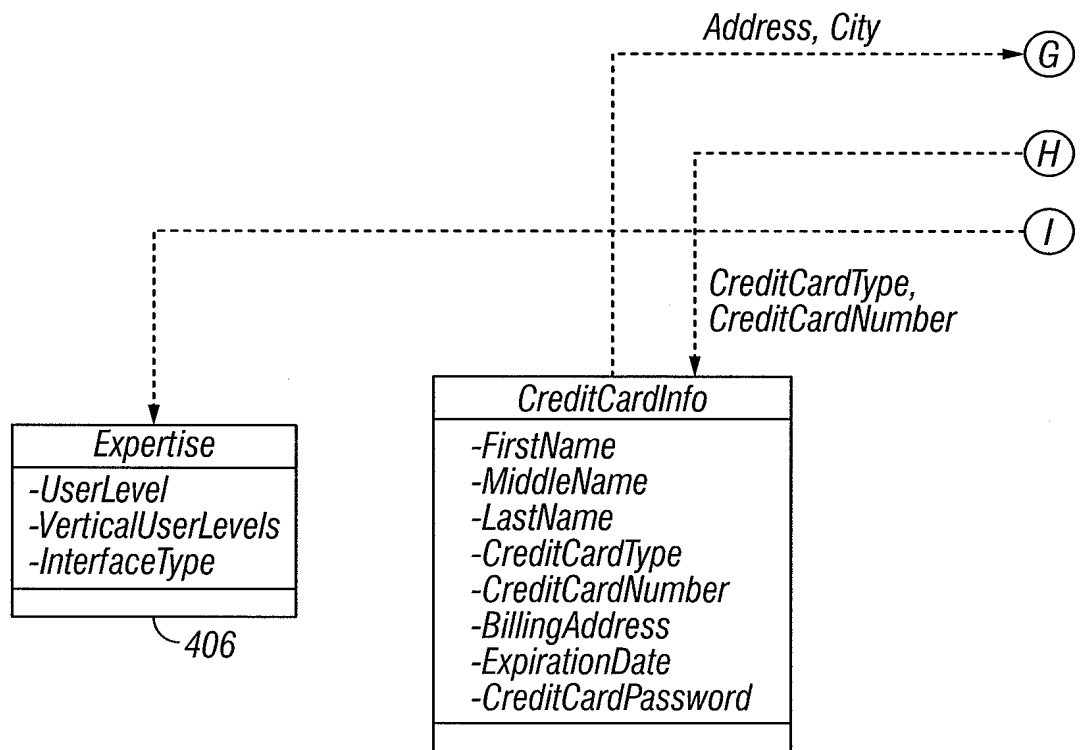
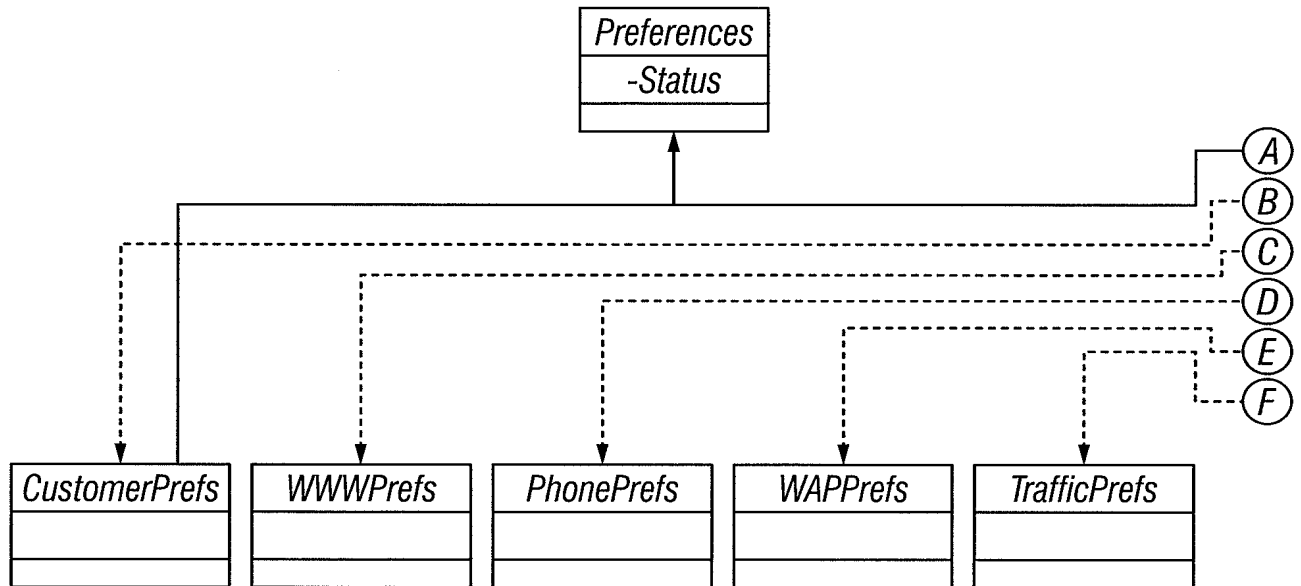


FIG. 4

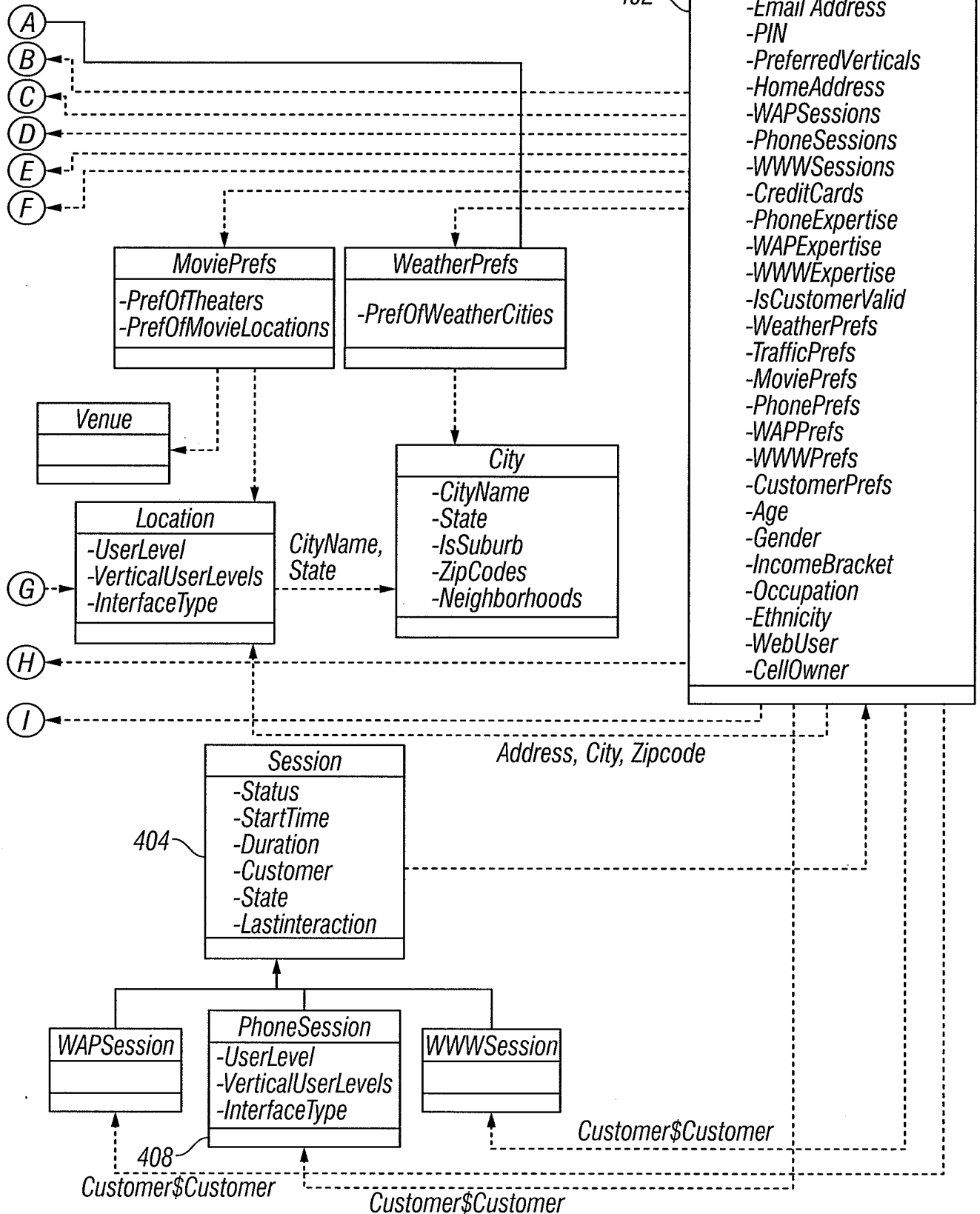
5/40

300

FIG. 4  
(Cont'd)

**6/40****FIG. 5**

**400**



8/40

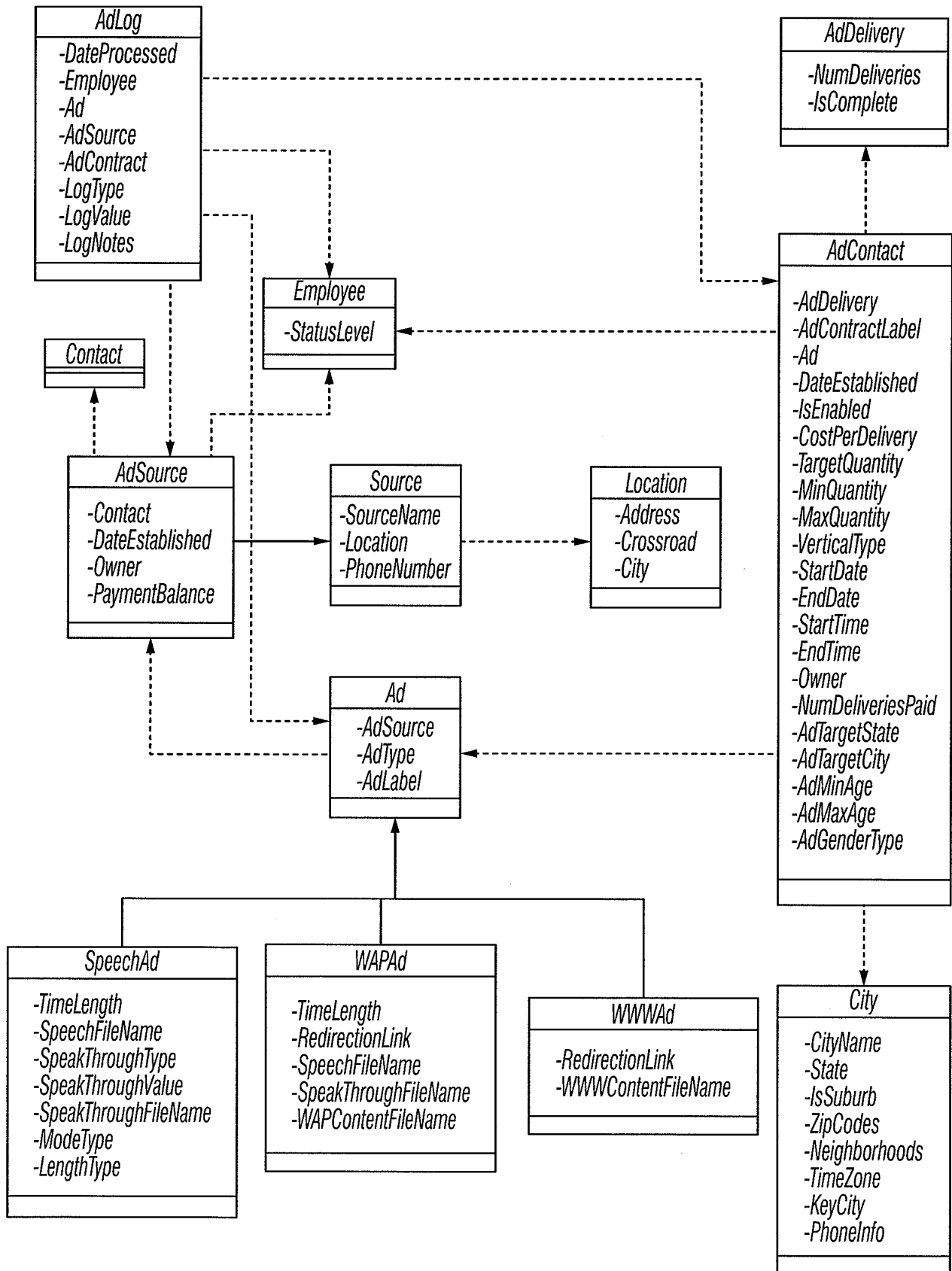
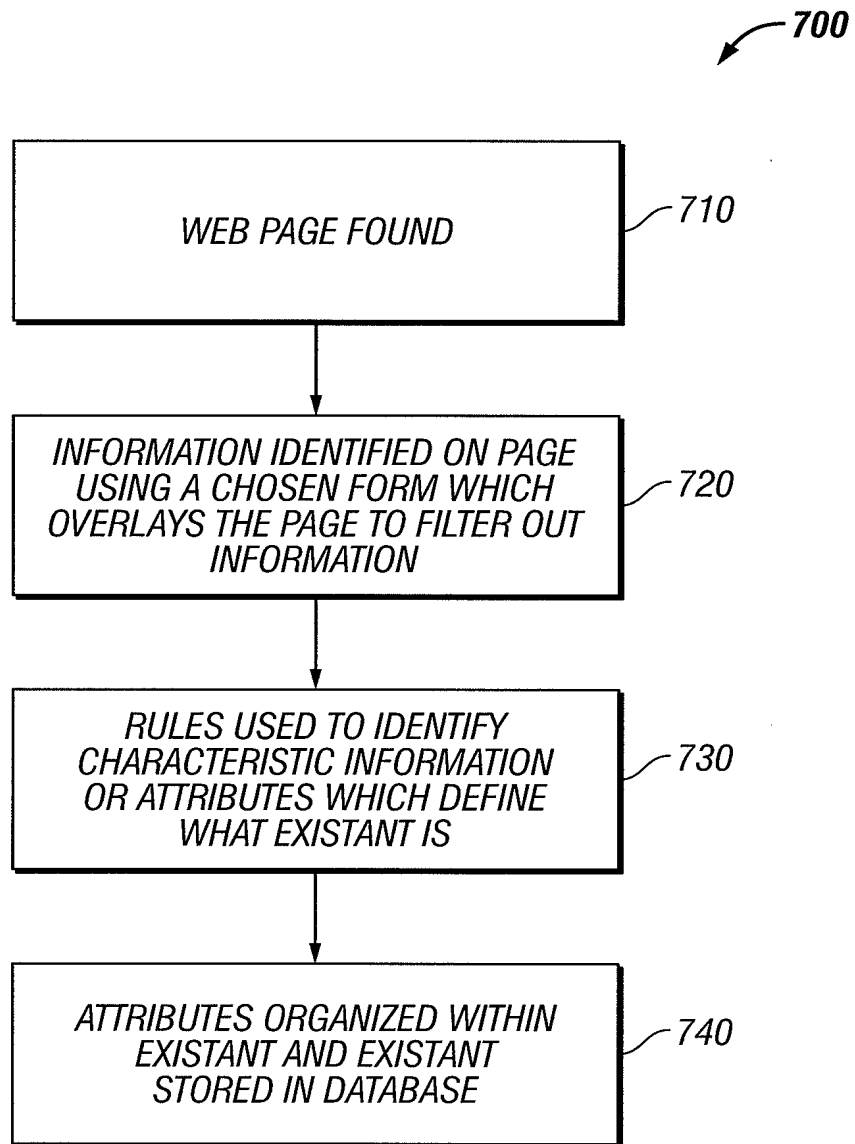


FIG. 6



**9/40**



**FIG. 7**

10/40

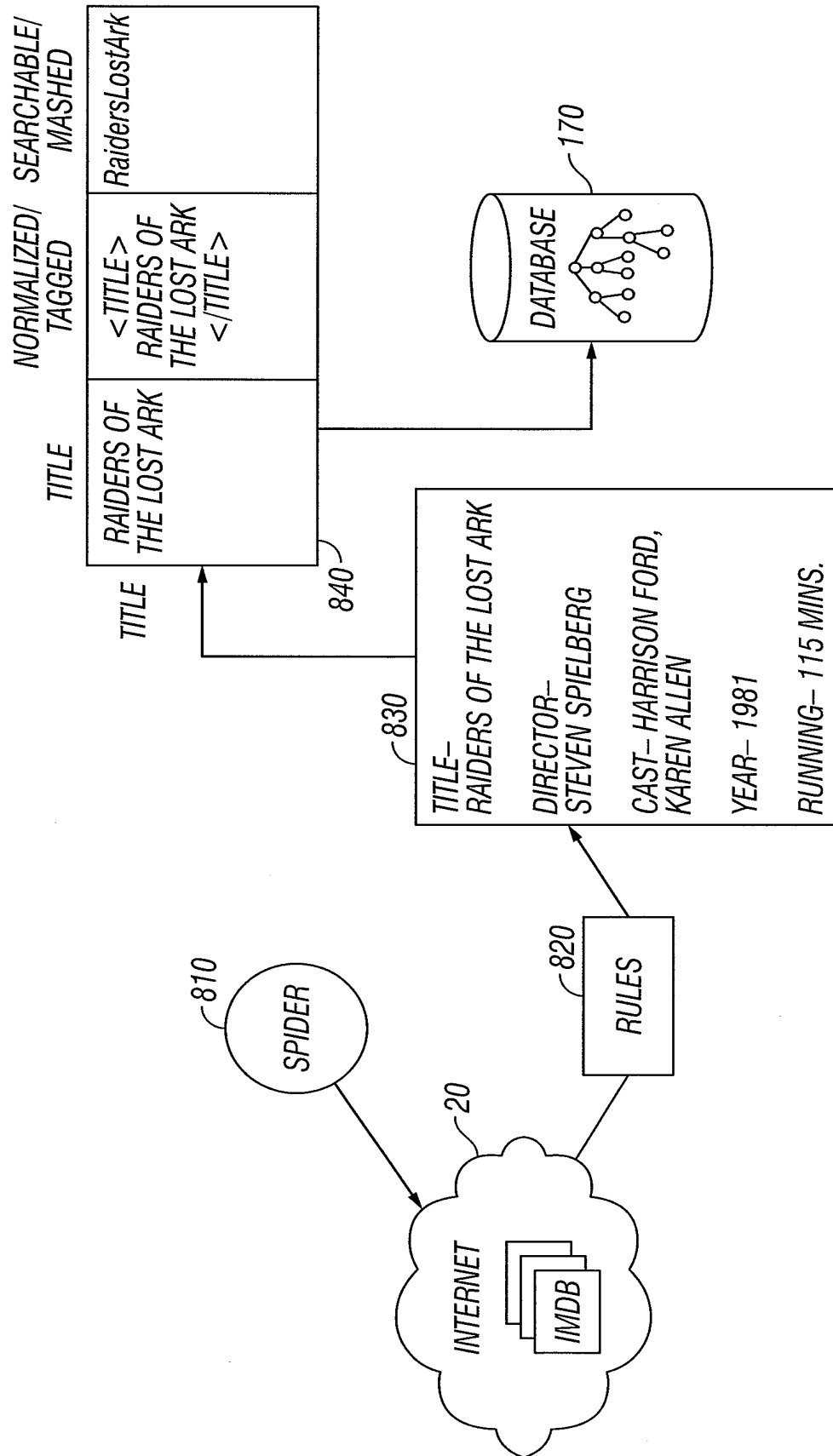
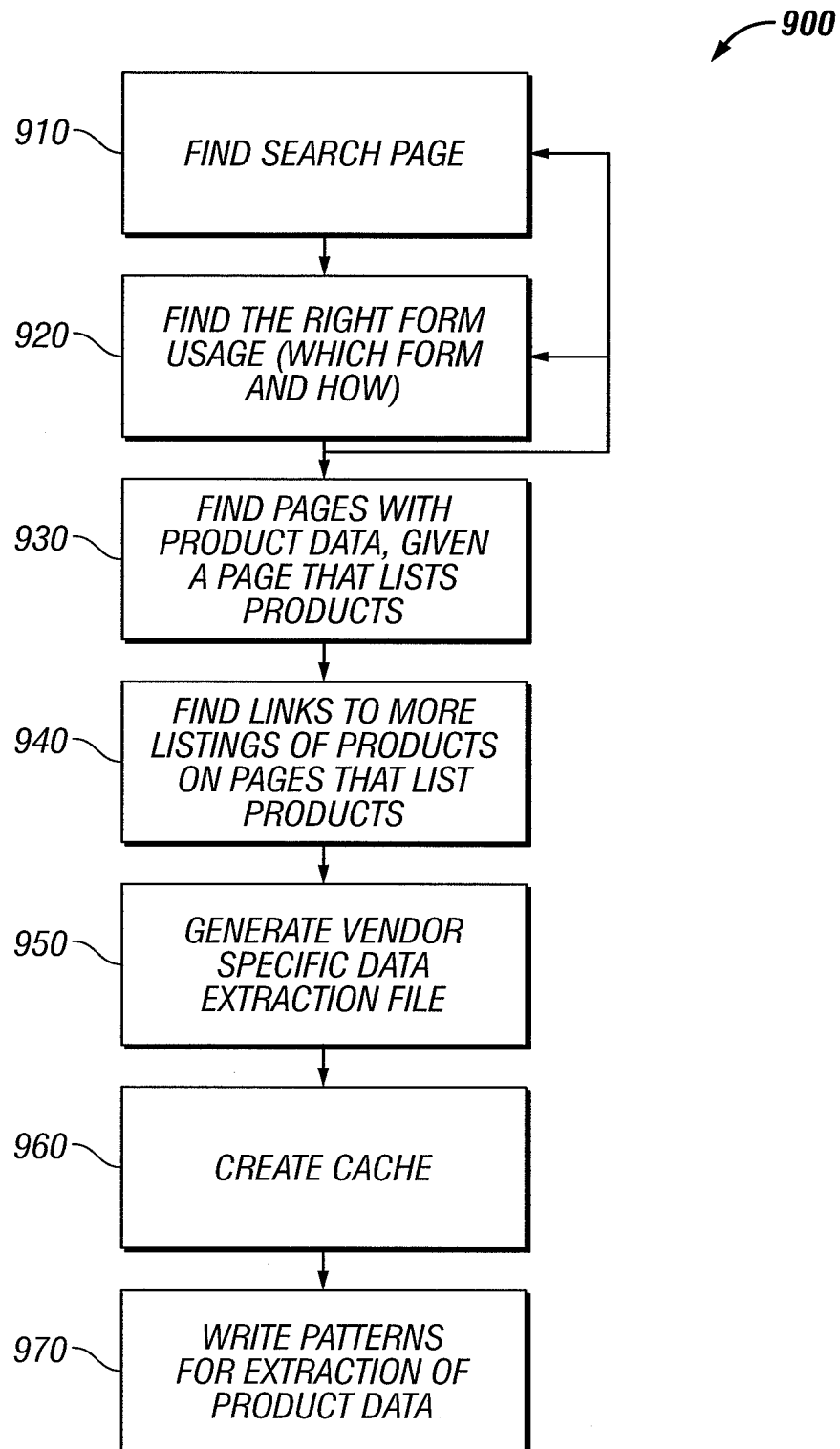
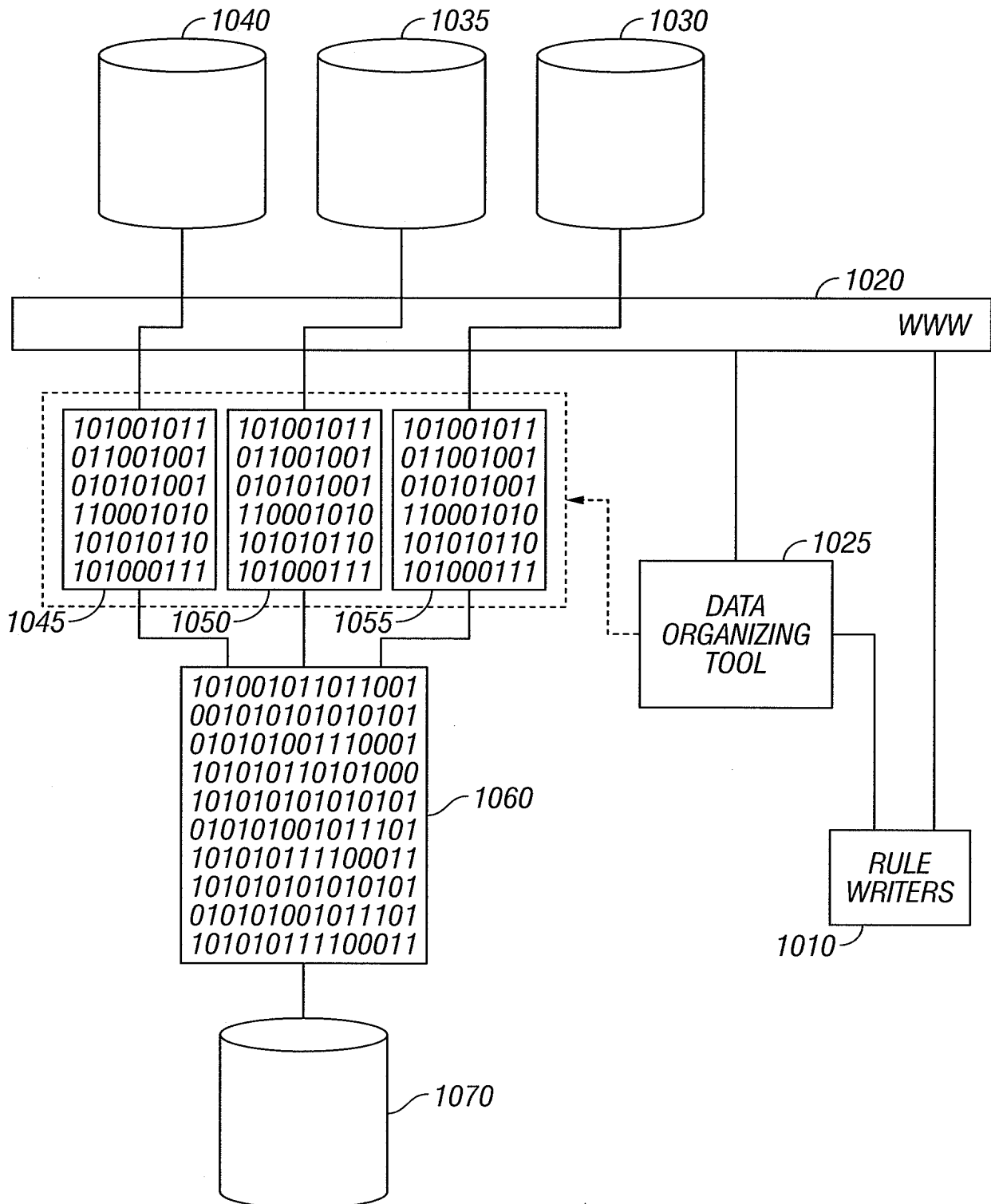
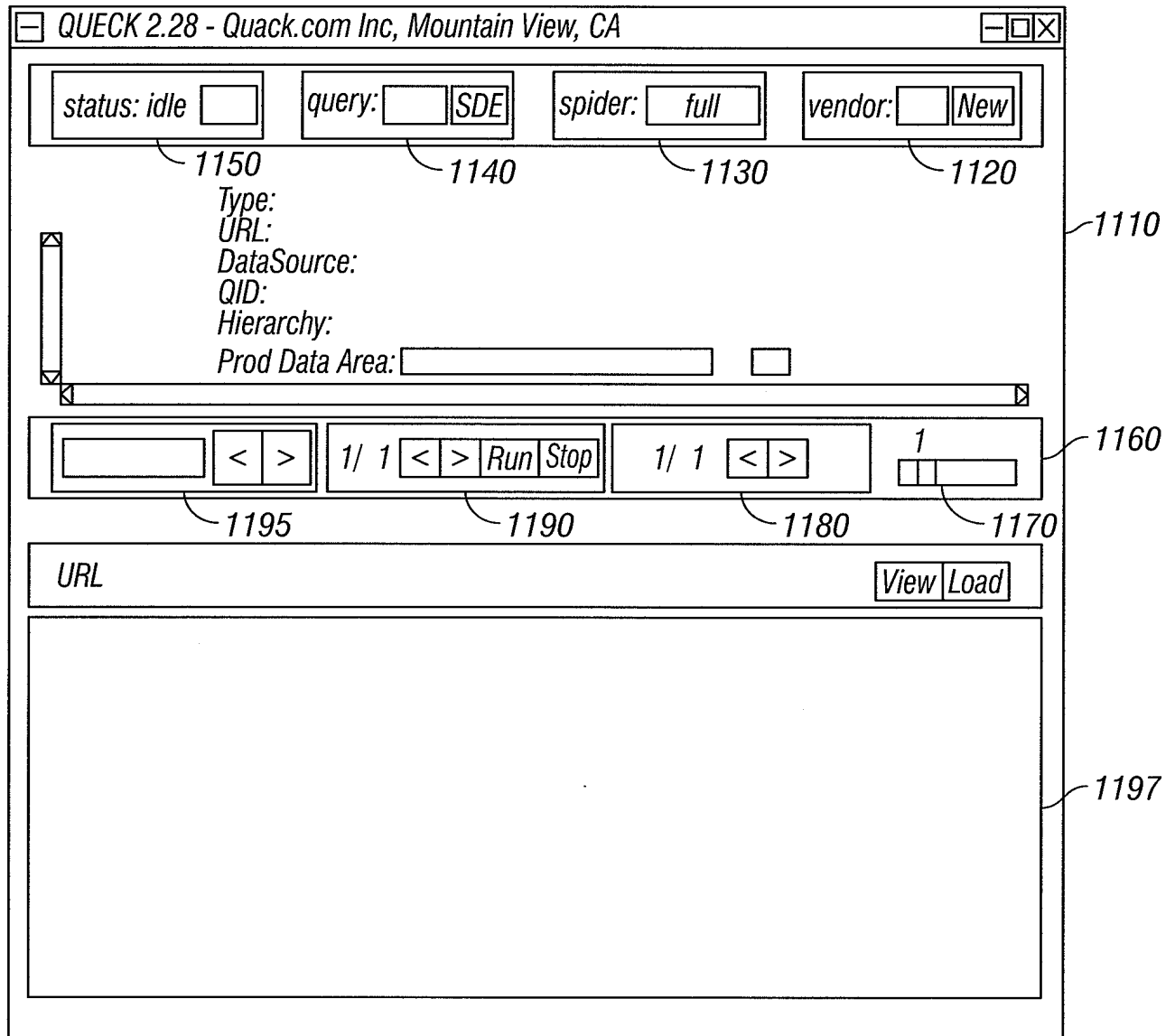


FIG. 8

**11/40****FIG. 9**

**12/40****FIG. 10**

**13/40****FIG. 11**

**14/40****1200** →

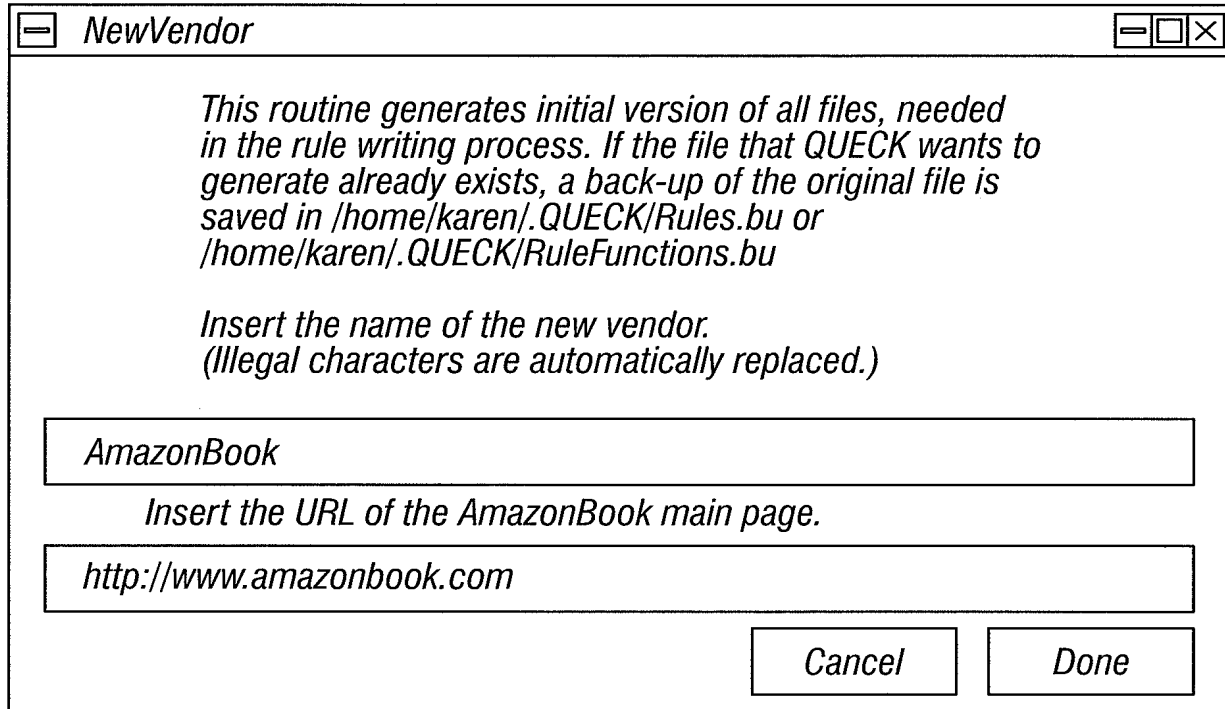
*This routine generates initial version of all files, needed in the rule writing process. If the file that QUECK wants to generate already exists, a back-up of the original file is saved in /home/karen/.QUECK/Rules.bu or /home/karen/.QUECK/RuleFunctions.bu*

*Insert the name of the new vendor.  
(Illegal characters are automatically replaced.)*

AmazonBook

Cancel Done

**FIG. 12**

**15/40****1300** →

**NewVendor**

*This routine generates initial version of all files, needed in the rule writing process. If the file that QUECK wants to generate already exists, a back-up of the original file is saved in /home/karen/.QUECK/Rules.bu or /home/karen/.QUECK/RuleFunctions.bu*

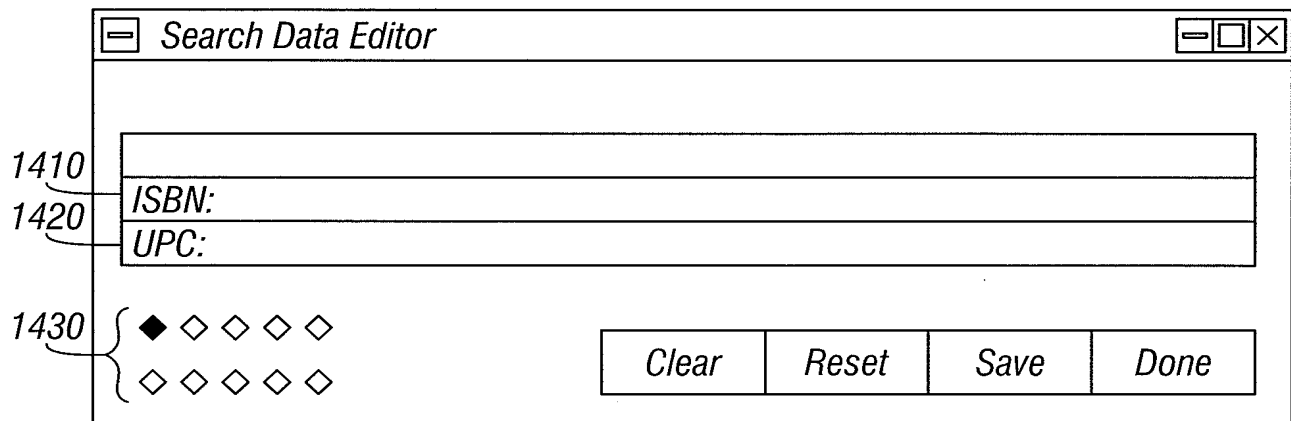
*Insert the name of the new vendor.  
(Illegal characters are automatically replaced.)*

AmazonBook

*Insert the URL of the AmazonBook main page.*

<http://www.amazonbook.com>

Cancel Done

**FIG. 13****1400** →

**Search Data Editor**

1410

1420 ISBN:

UPC:

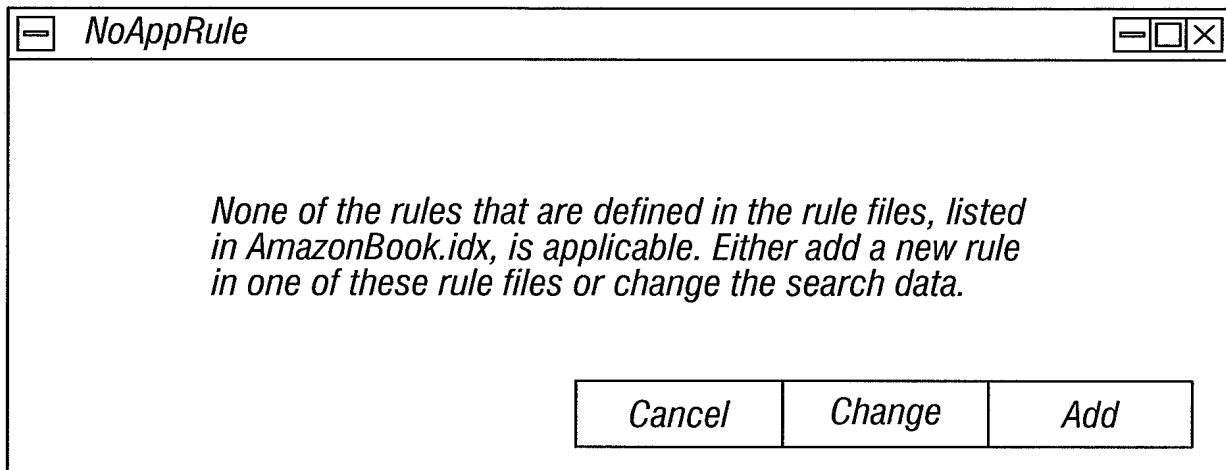
1430 { ◆ ◇ ◇ ◇ ◇  
◇ ◇ ◇ ◇ ◇

Clear Reset Save Done

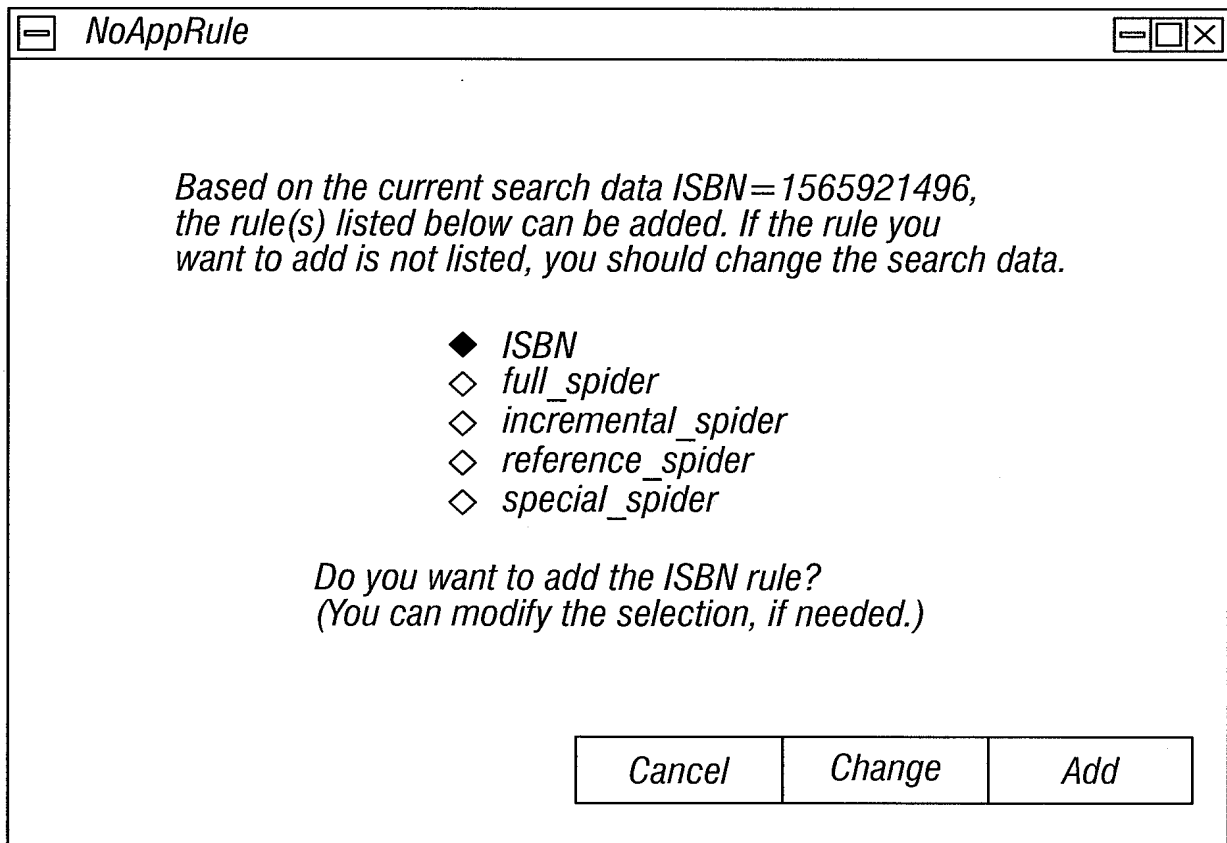
**FIG. 14**

**16/40**

1500

**FIG. 15**

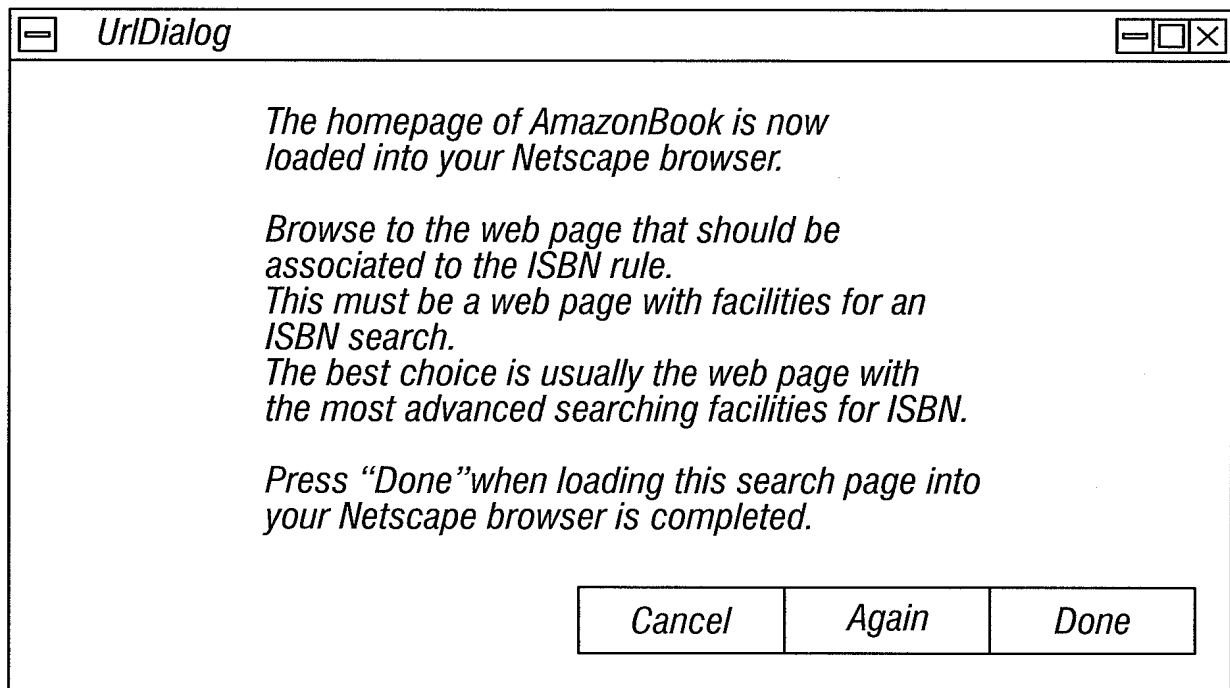
1600

**FIG. 16**

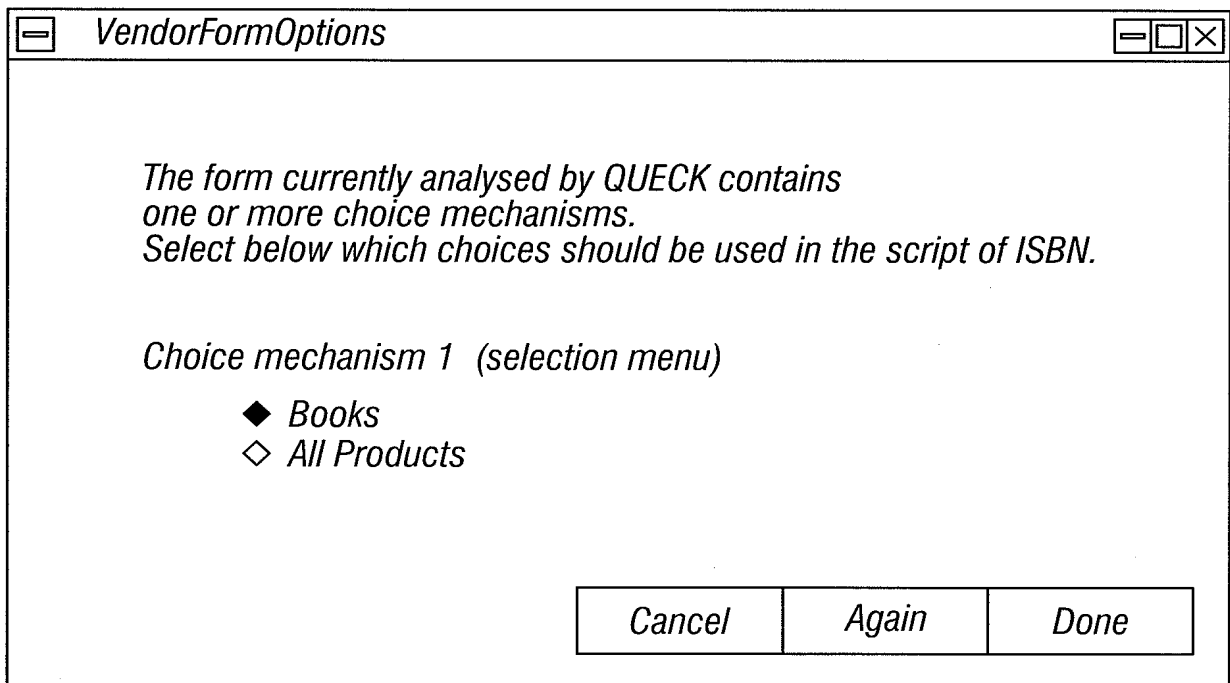


**17/40**

1700

**FIG. 17**

1800

**FIG. 18**

**18/40**

1900 →

*QUECK has computed the URL of the page corresponding to your query ISBN=1565921496.*

*The computation is based on the first form of the search page.*

*The result is loaded into Netscape. If the resulting page is incorrect, press "Next" to analyze the next form on the search page.*

*If the resulting page is correct, press "OK".*

Cancel      Next      OK

**FIG. 19**

2000 →

*Insert the pattern that must be used to detect single products on pages that list multiple products. Use the contents of the editor to develop and test your pattern.*

*Your pattern must set \$1 to the URL of the single product. QUECK is smart enough to prefix this with <http://www.amazon.com> in case that is missing.*

*If moreover \$2 is set to string that identifies the single product, this string will be used in debugging and logging information. Setting \$2 is not required however.*



     . \*      first

Defer      Count      Match      Done

**FIG. 20**

**19/40**

2100

 NSP 

*Insert the pattern that must be used to detect links on multiple products pages to even more multiple product pages. Use the "Match" button to test your pattern.*

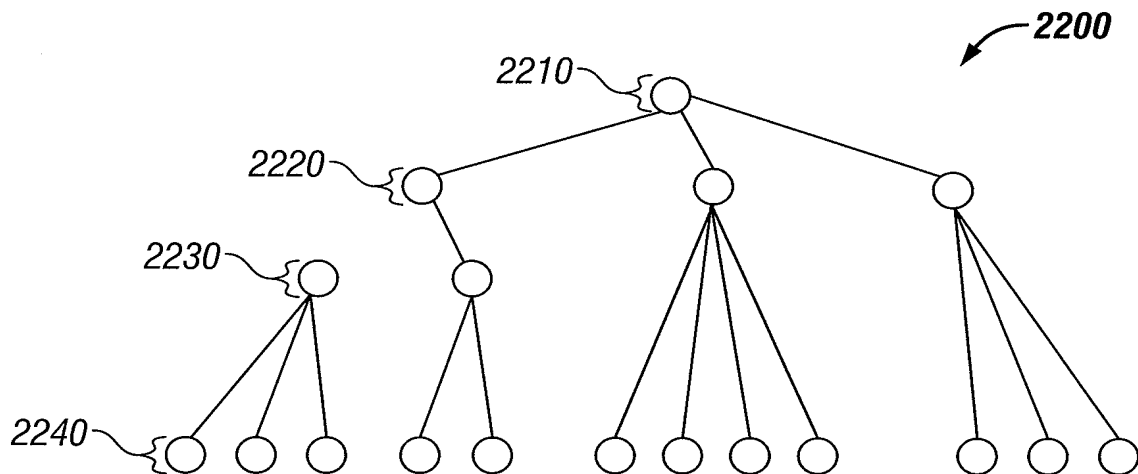
*Your Pattern must set \$1 to the URL of the new multiple product page. QUECK is smart enough to prefix this with <http://www.amazon.com> in case that is missing.*

*If your query does not generate enough product hits to have more than one multiple product page, you can choose "Defer" and defer the configuration until you run a query that actually does generate enough product hits to have more than one multiple product page.*

Defer

Match

Build

**FIG. 21****FIG. 22**

2300

20/40

*Insert here the URL of the page, currently loaded into Netscape. This is the page associated to the full\_spider rule.*

*Next, set "SpiderDepth" to the maximum number of links that has to be followed from the top of the hierarchy to the actual product pages. Note that in some cases this number depends on the branch you follow. Setting "SpiderDepth" too low creates a spider that misses products that are nested too deep in the hierarchy. Setting "SpiderDepth" too high leads to a decrease in performance.*

SpiderDepth: 1  
UpperBound: 0

Done

FIG. 23

2400

*The spider you specified is a level - 1 spider.  
This means that your spider has the following form:*

*level - 0: The top page (accessed via the URL above)  
level - 1: The single product pages to be spidered.*

*Insert below the pattern used to detect level - 1 pages on the top page.*

*Your pattern must set \$1 to the URLs of the child pages. QUECK is smart enough to prefix this URL with http://www.amazon.com in case it is missing. If your pattern also sets \$2, that value will be used in the hierarchy attributes.*

first 1st Level .\*

Cancel Count Match Build

FIG. 24

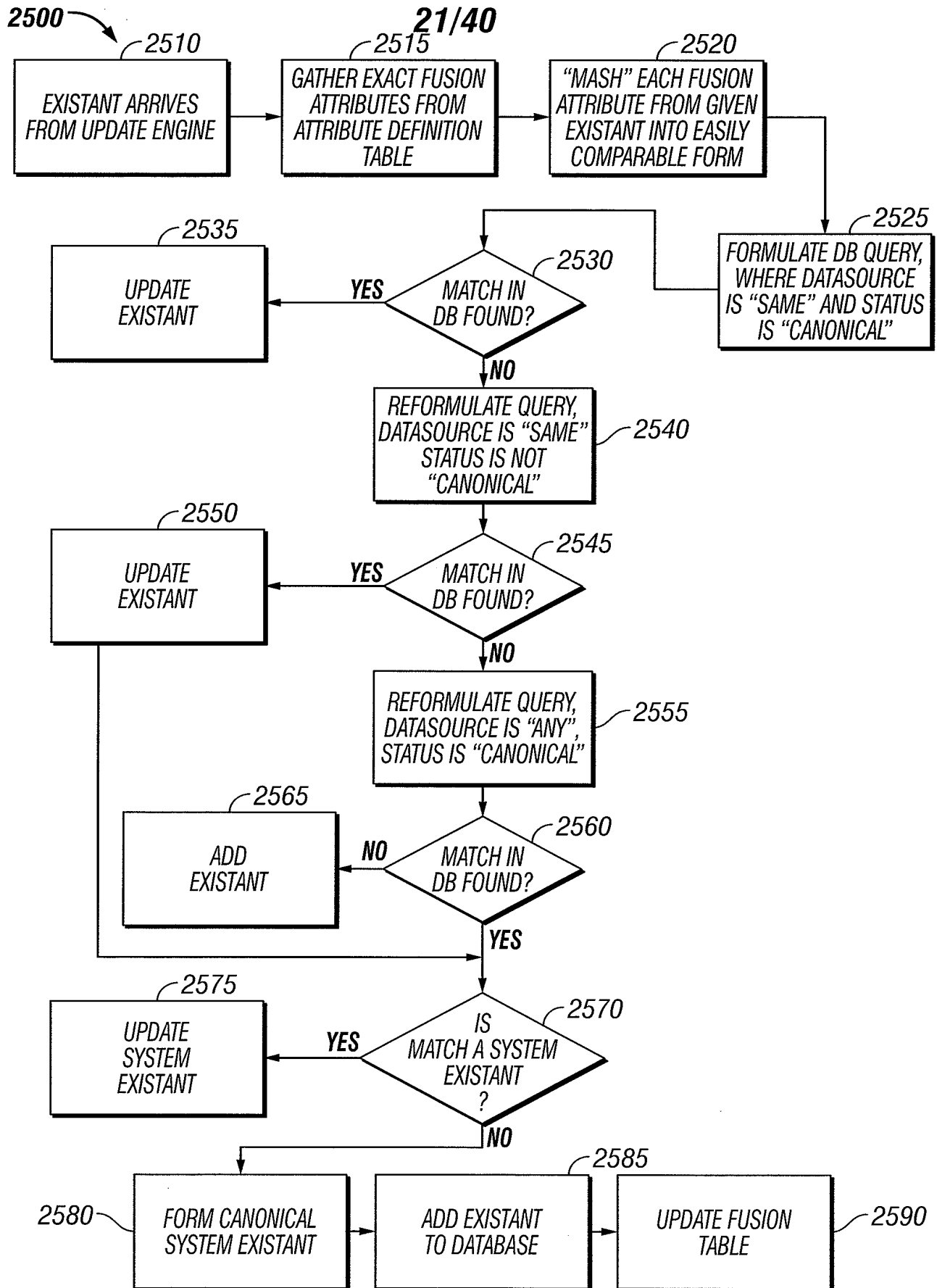


FIG. 25

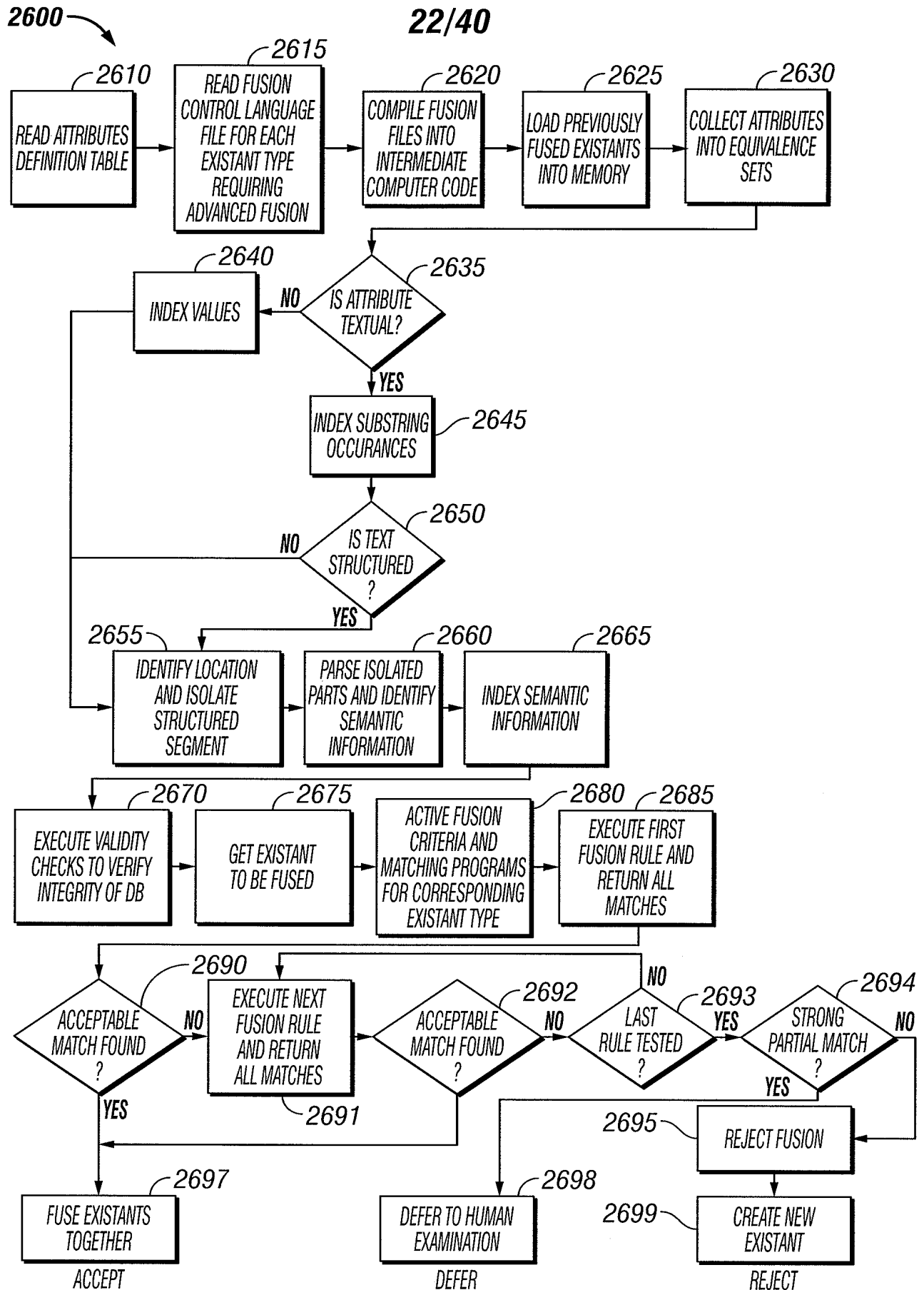
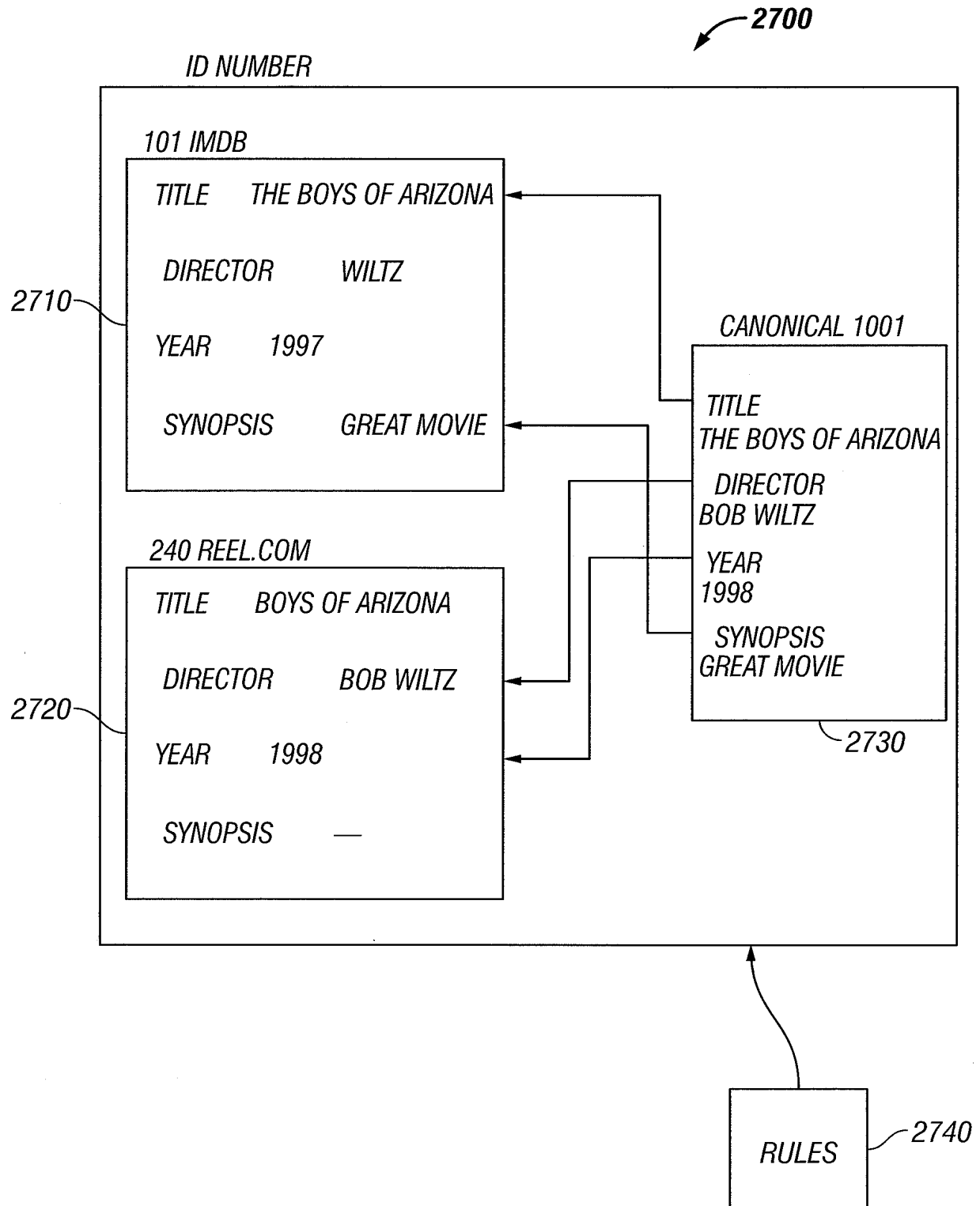


FIG. 26

**23/40****FIG. 27**

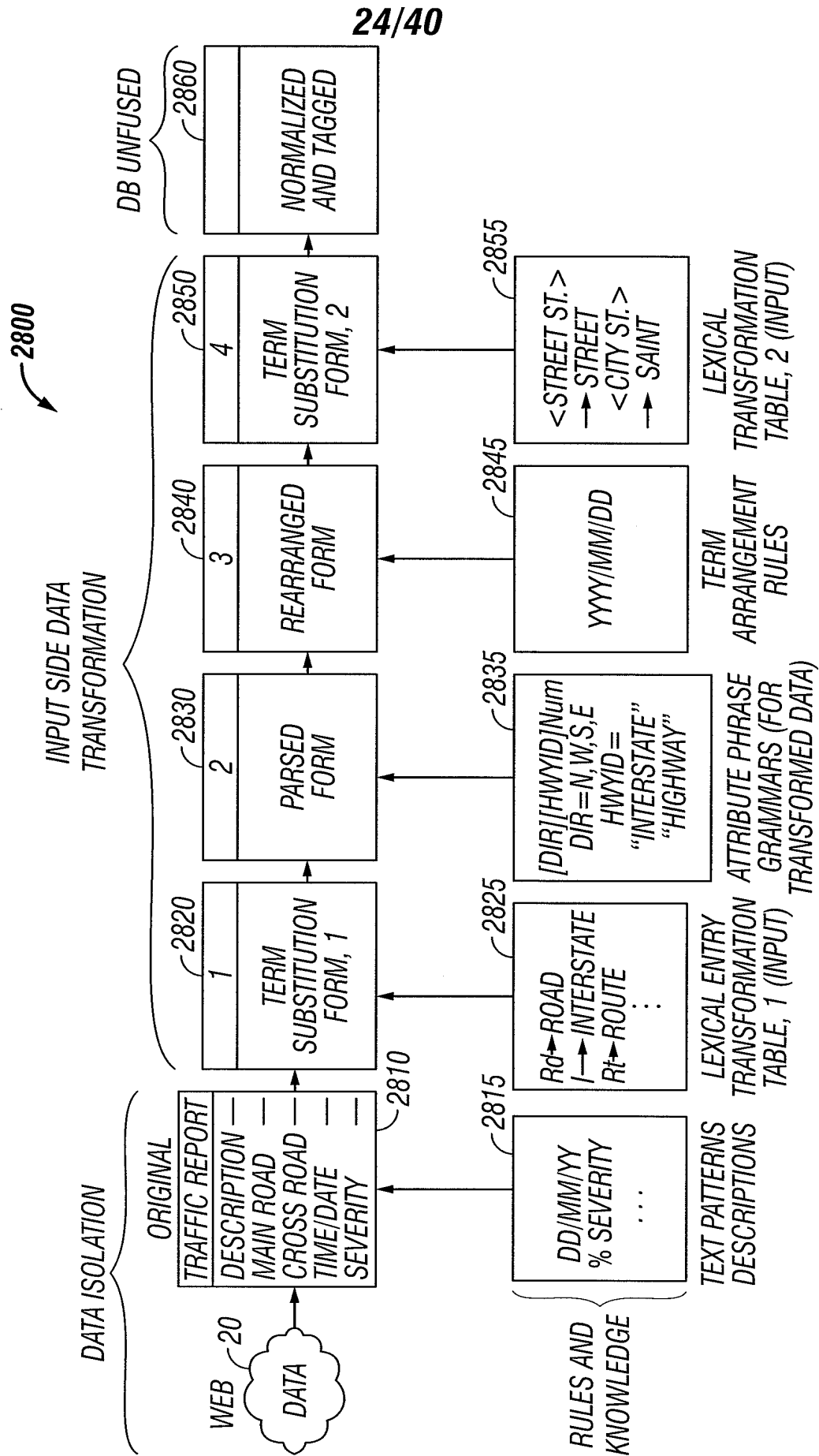


FIG. 28



25/40

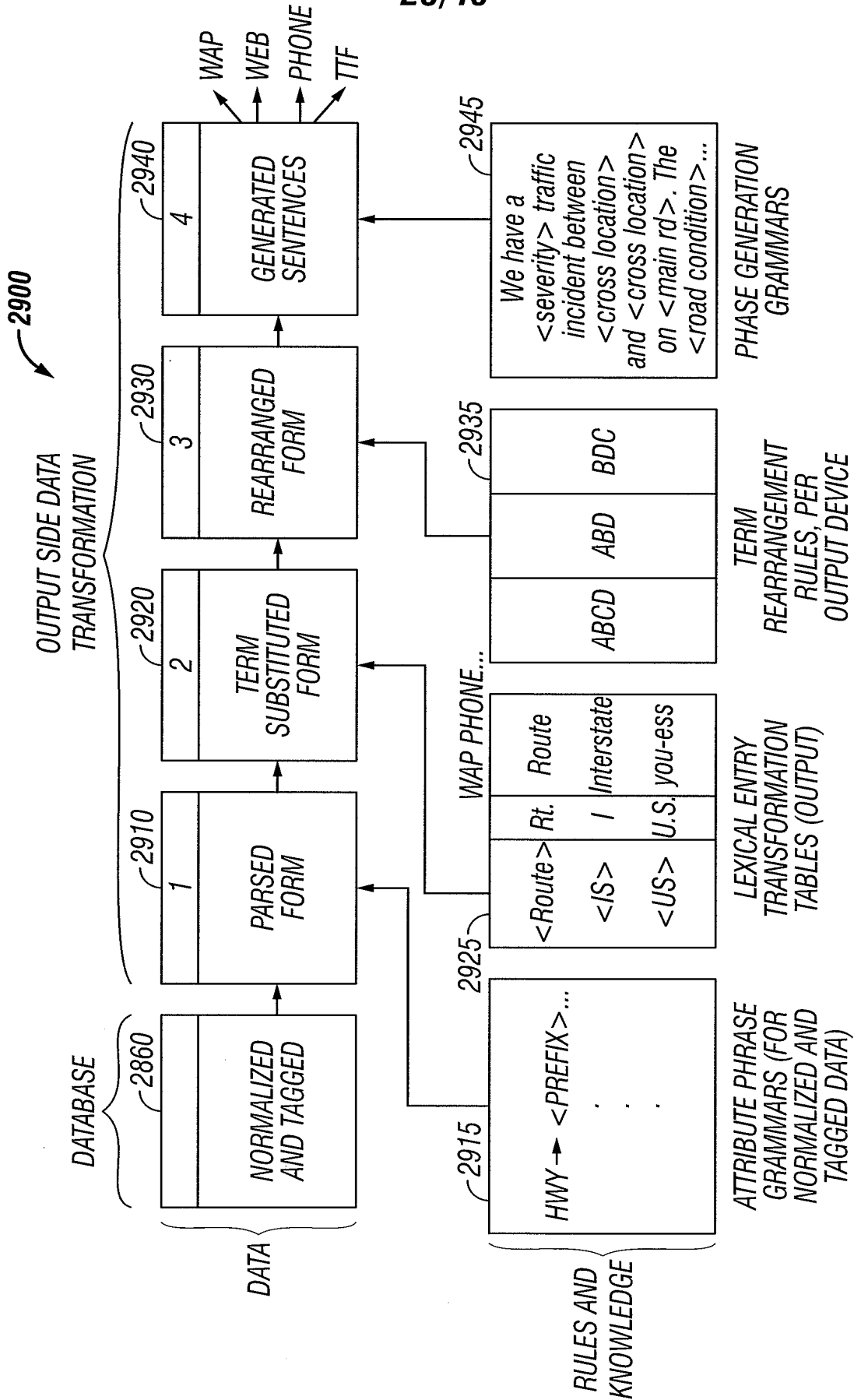
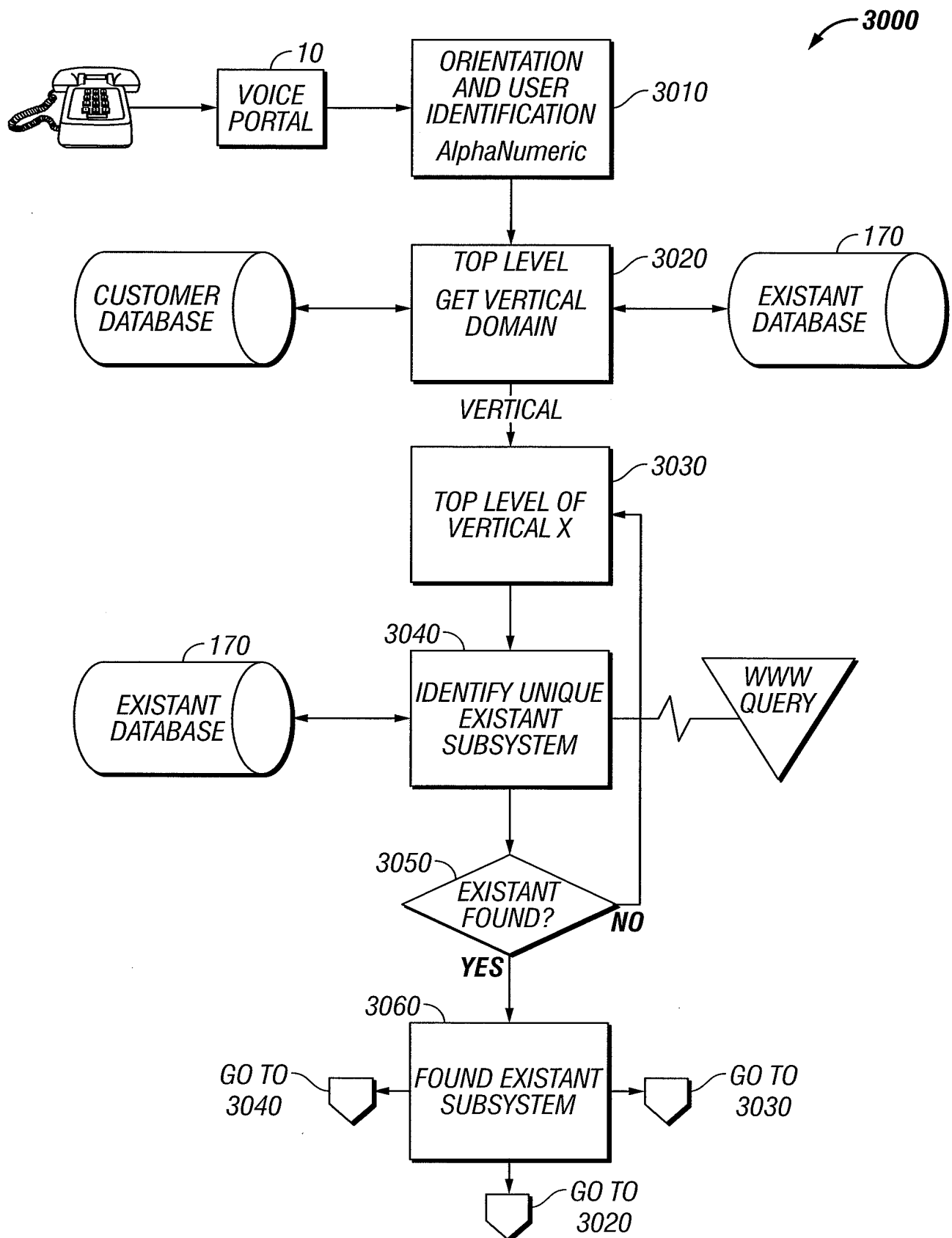
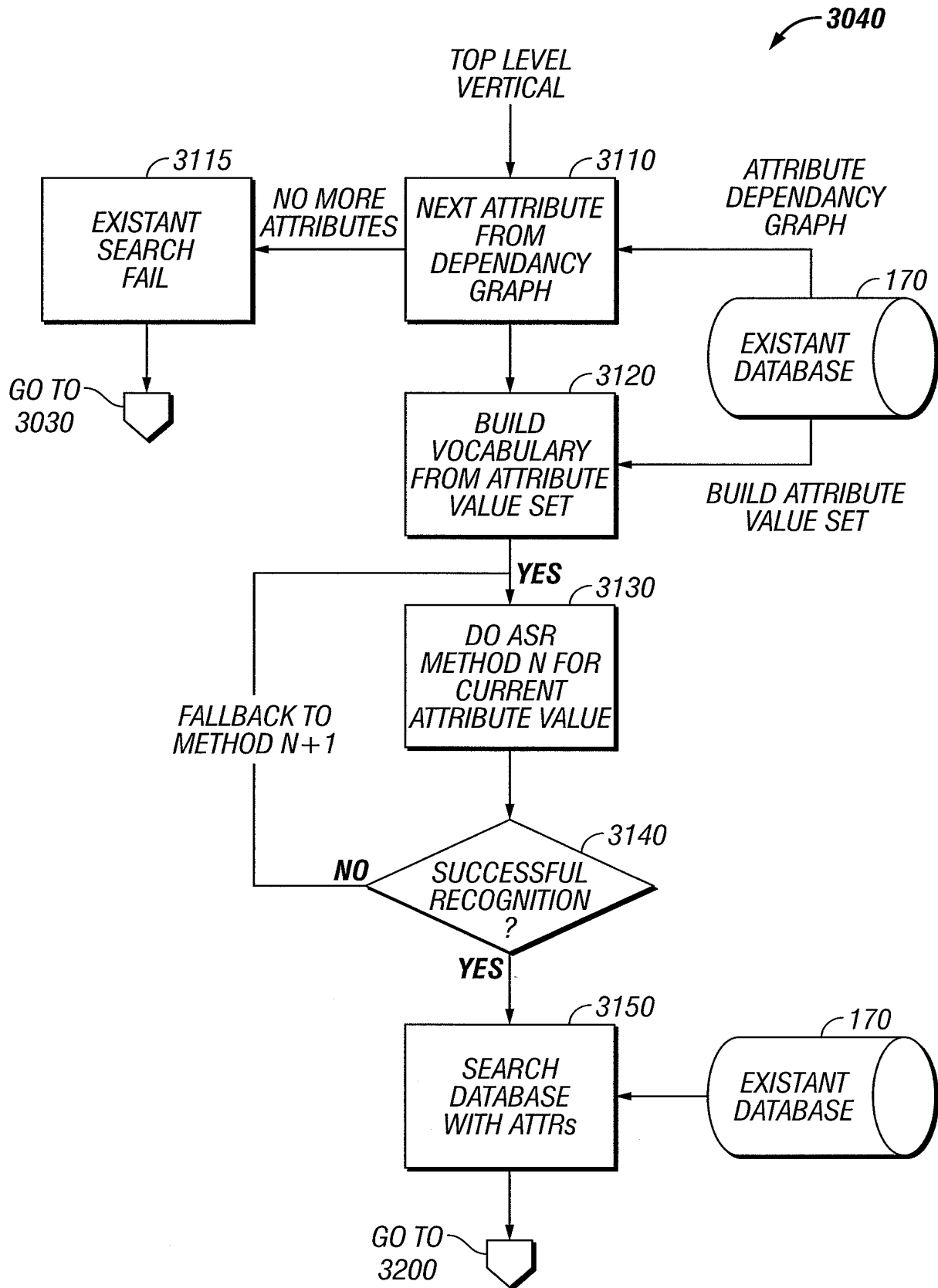


FIG. 29

**26/40****FIG. 30**

**27/40****FIG. 31**

28/40

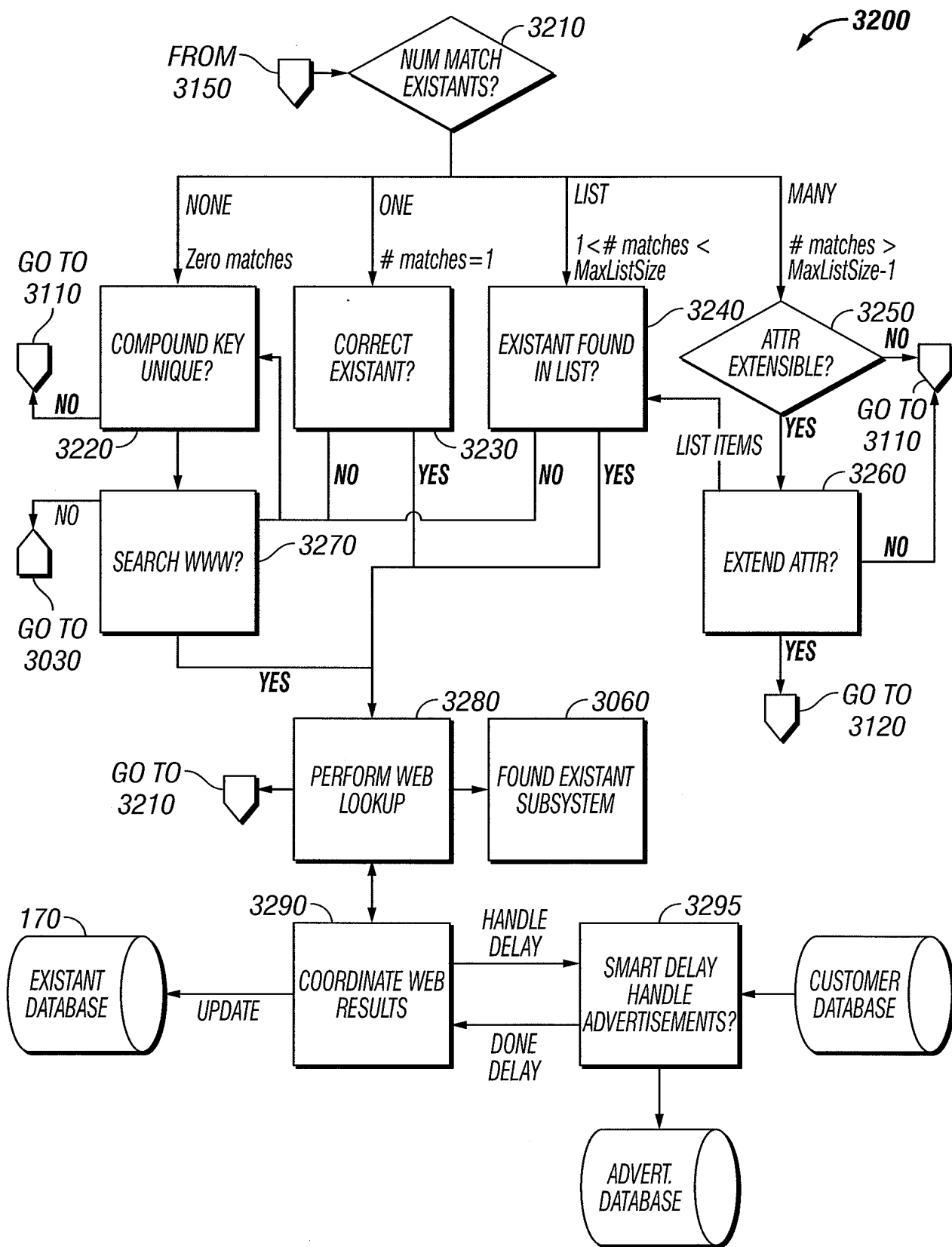
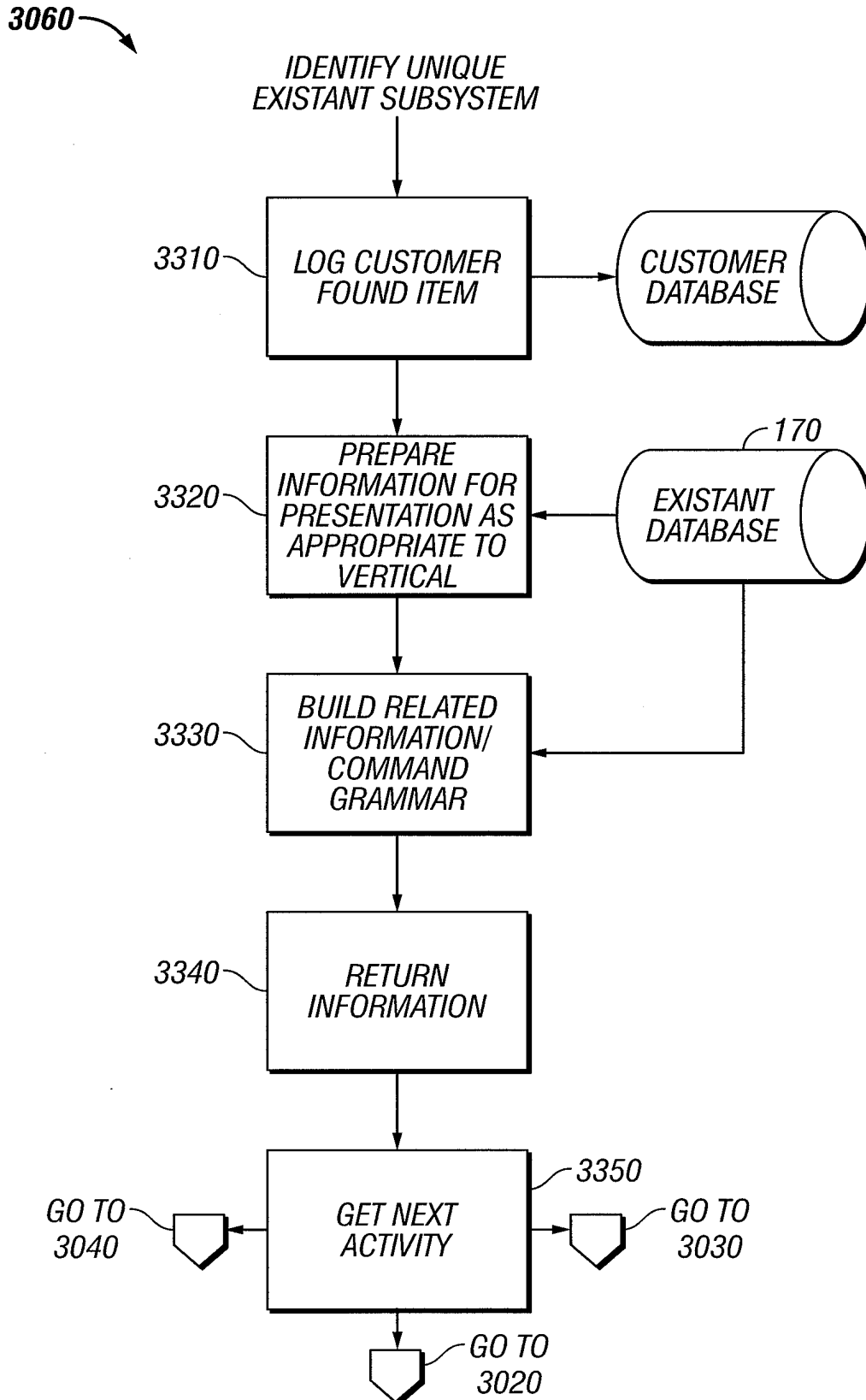


FIG. 32

**29/40**



**FIG. 33**

3400

30/40

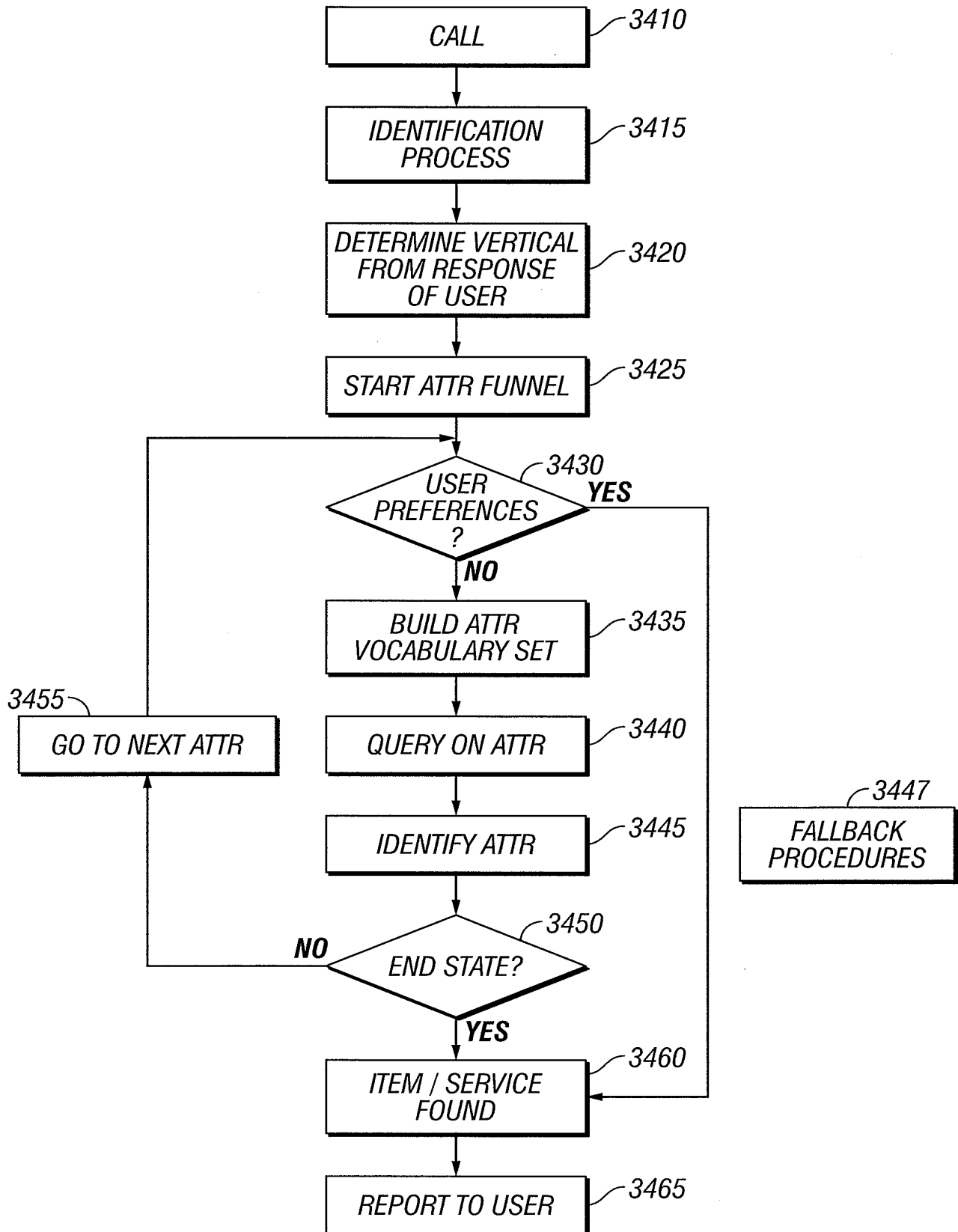
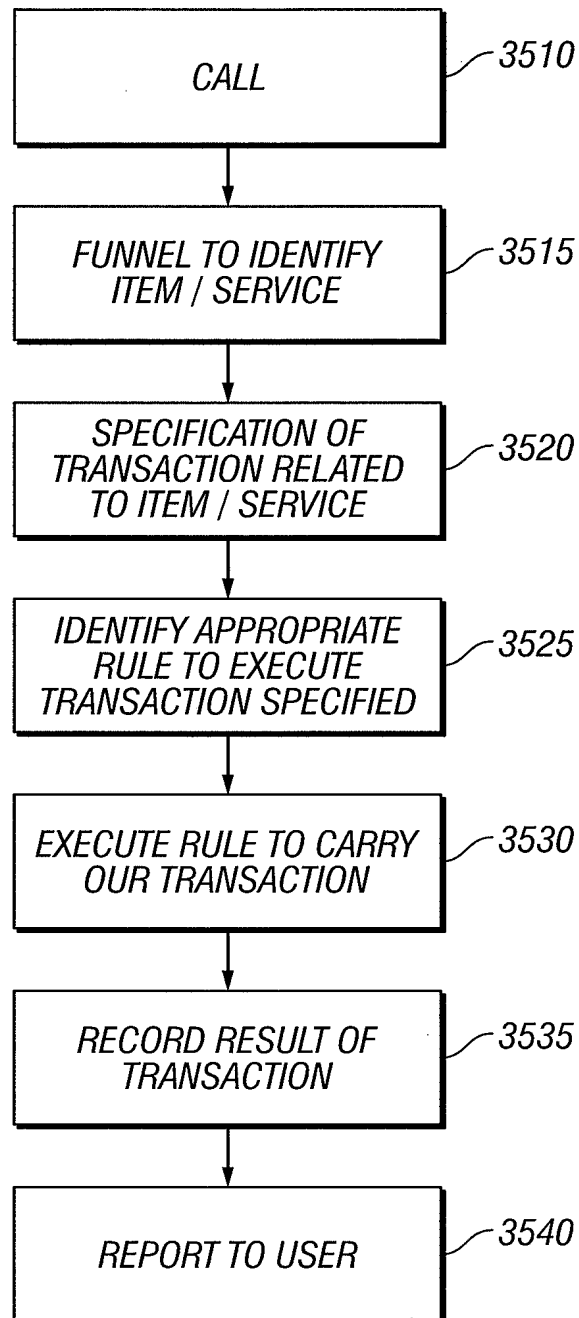
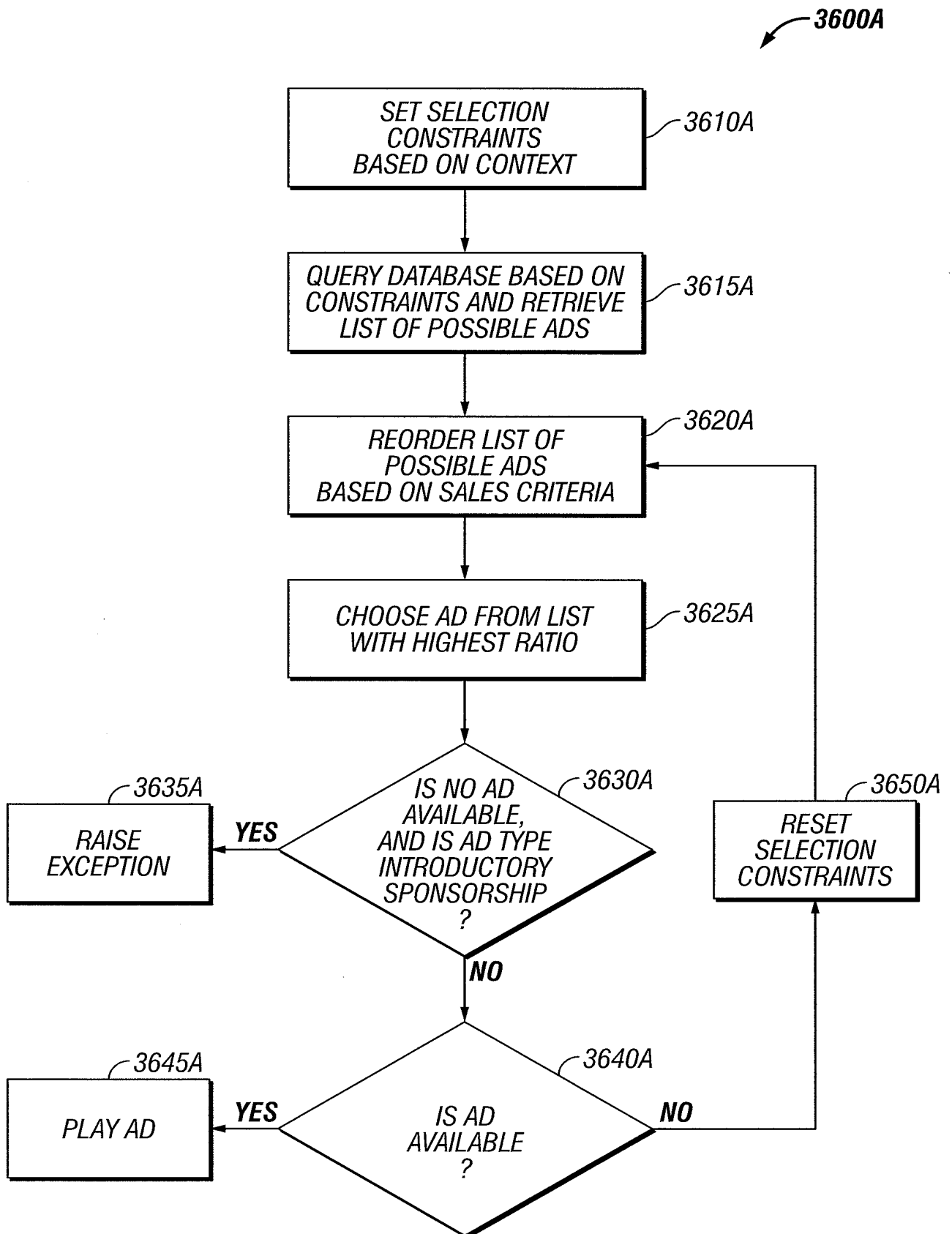
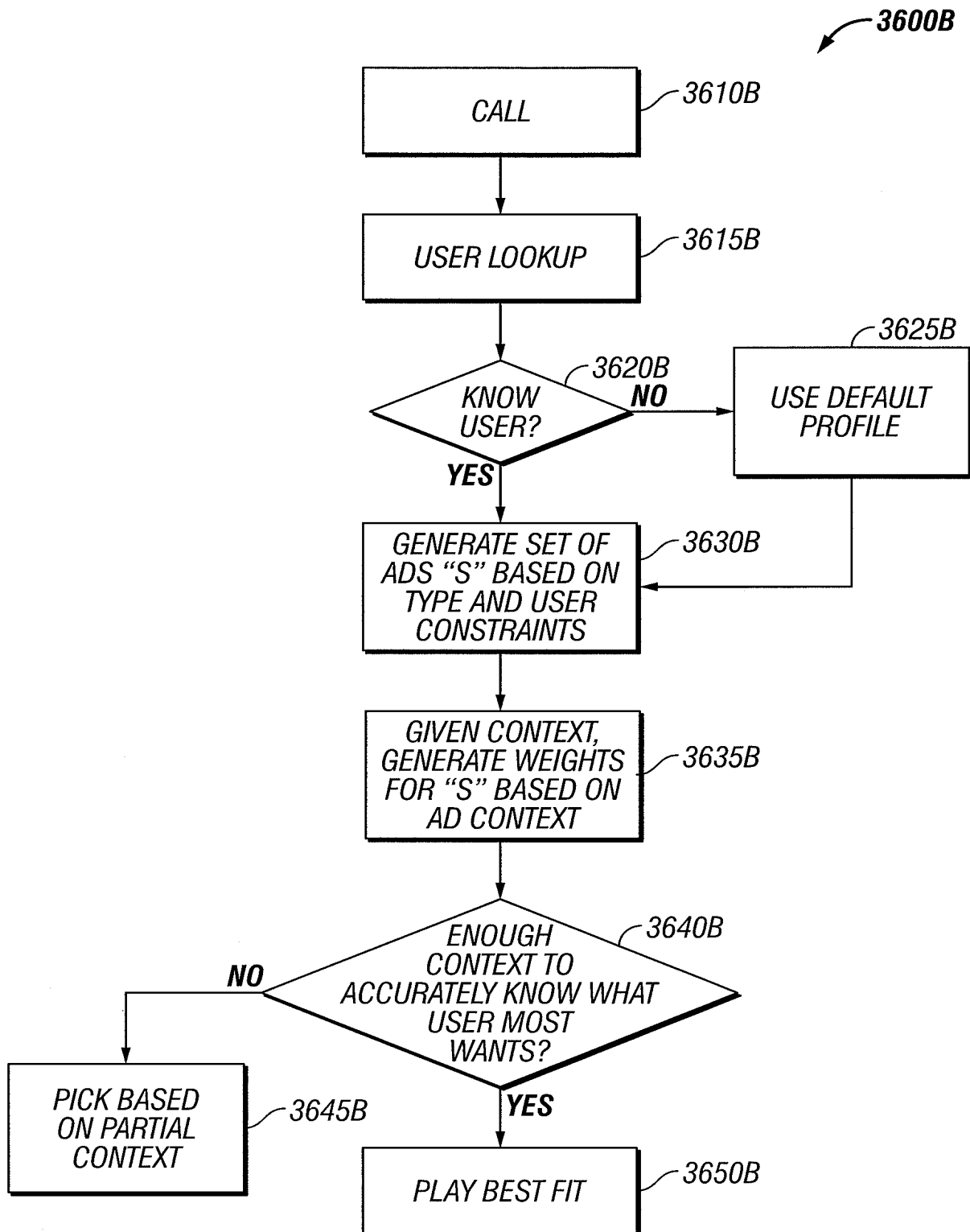


FIG. 34

**31/40****3500** →**FIG. 35**

**32/40****FIG. 36A**



**33/40****FIG. 36B**

34/40

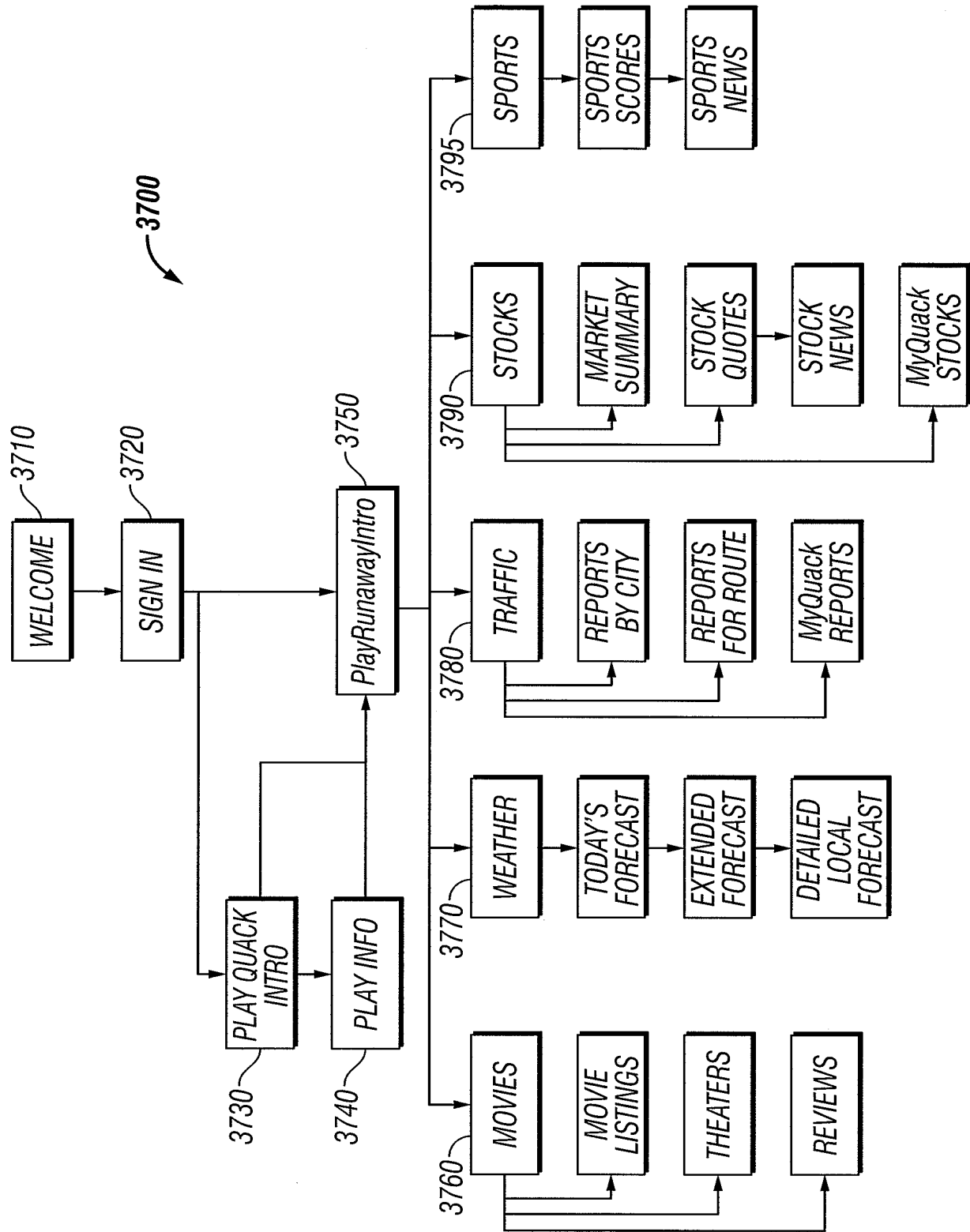
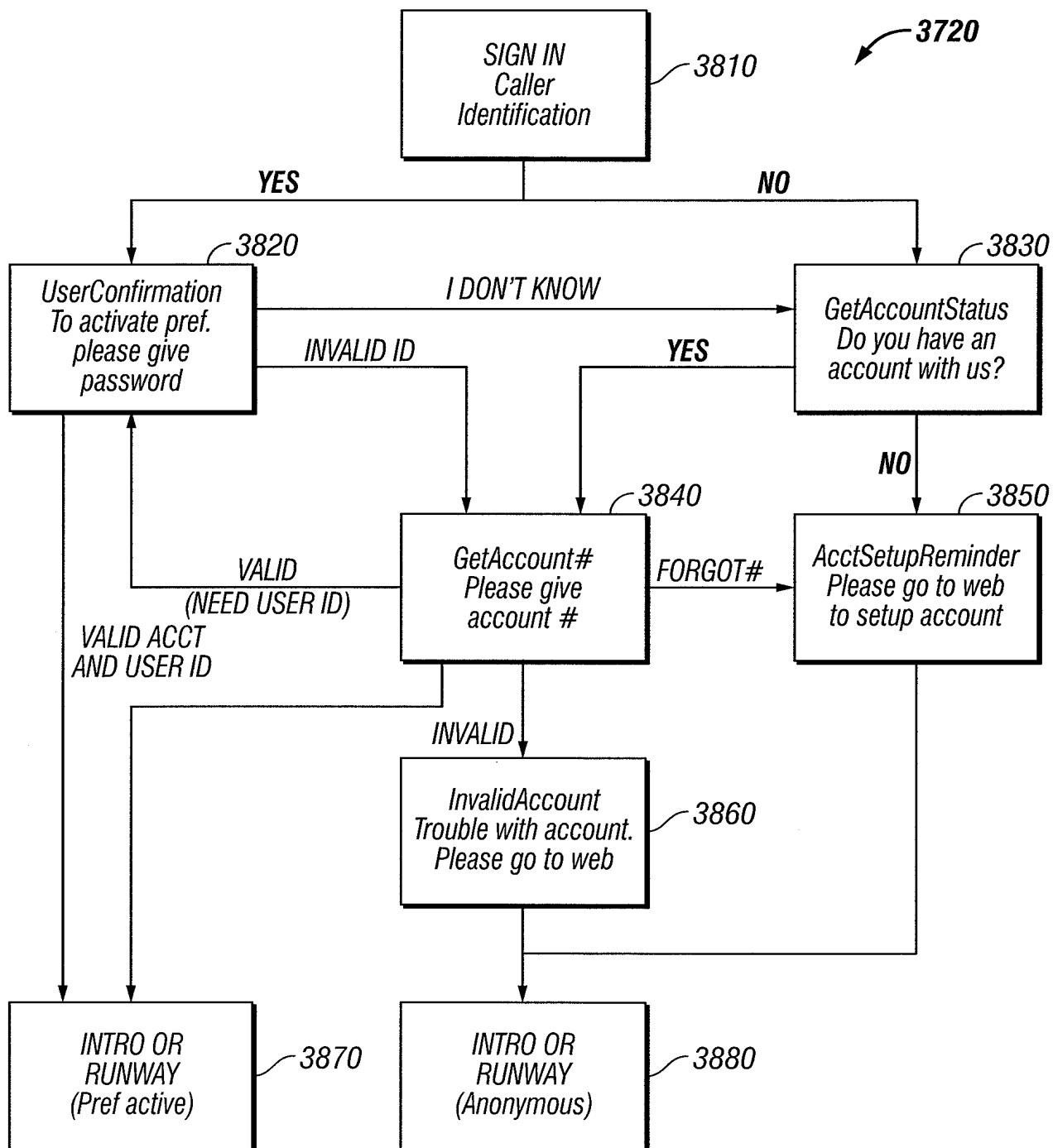
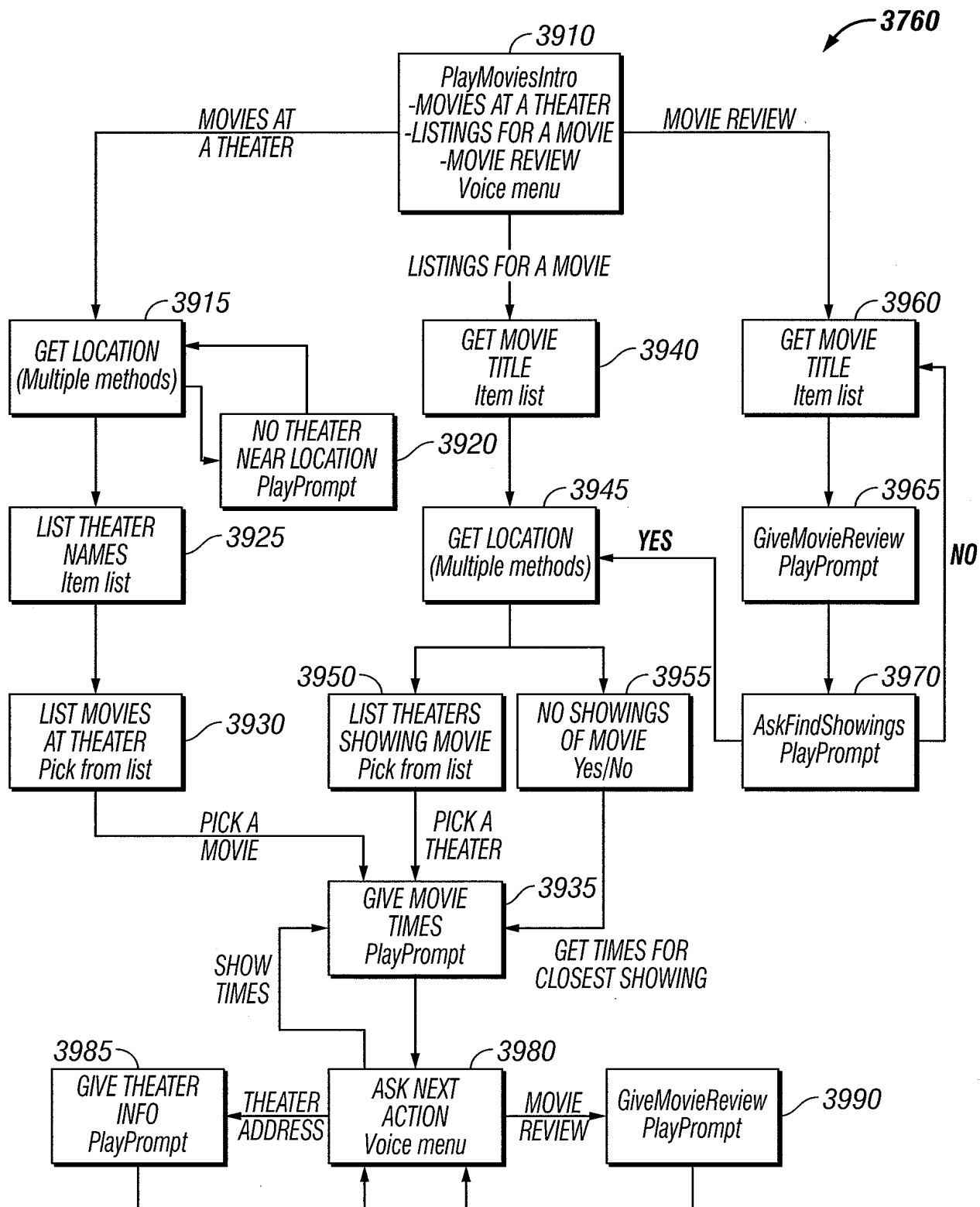
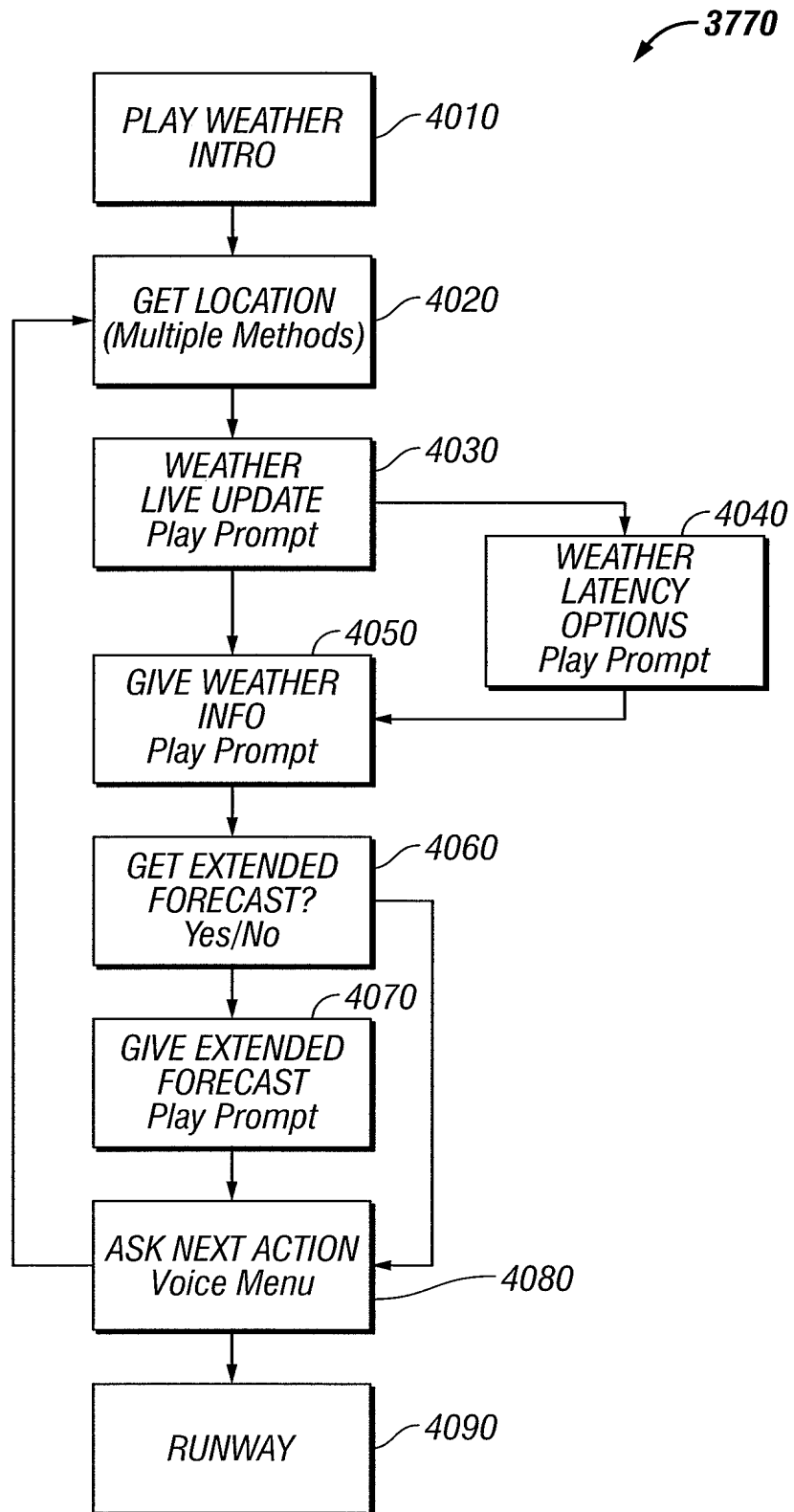
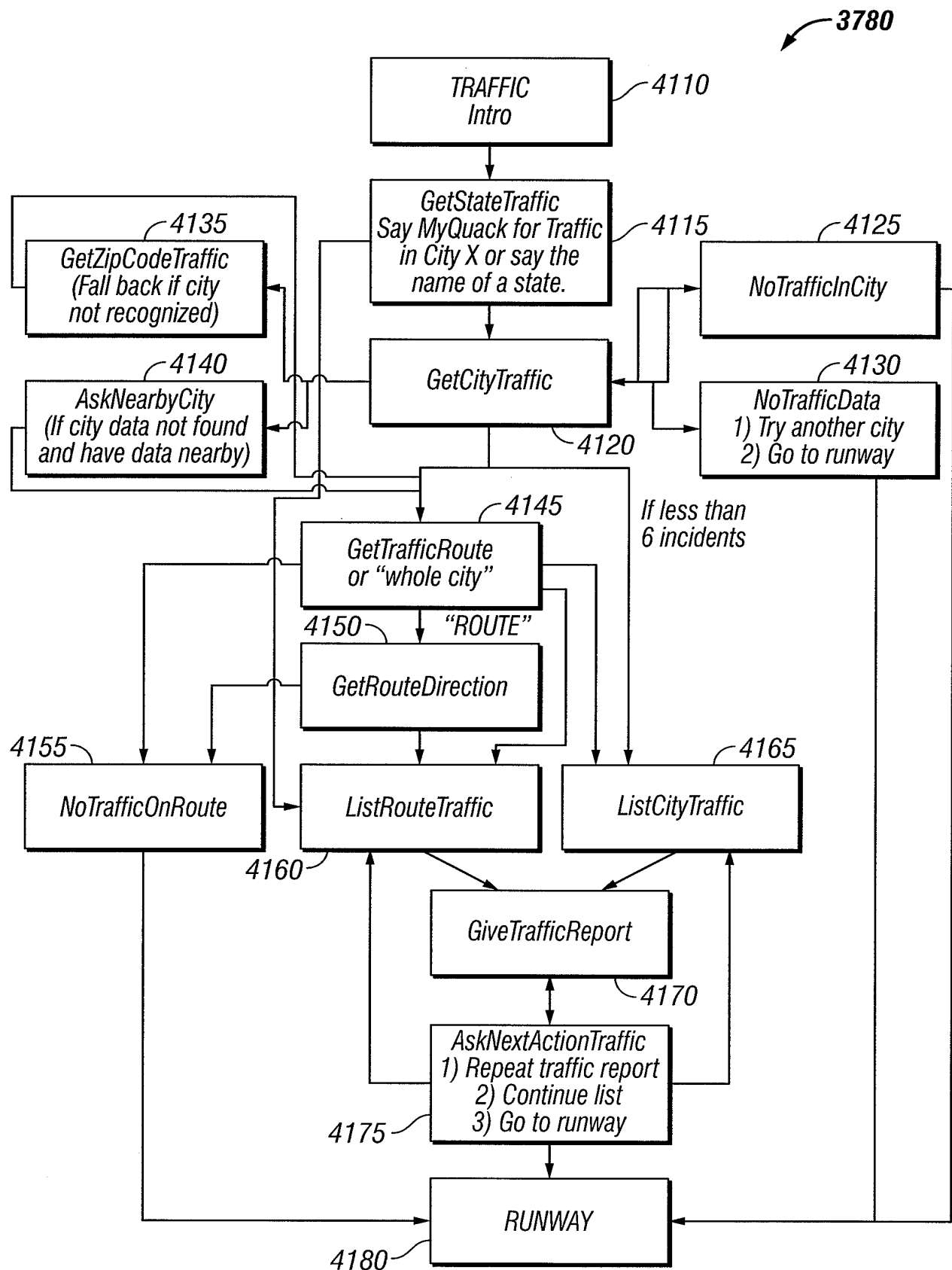


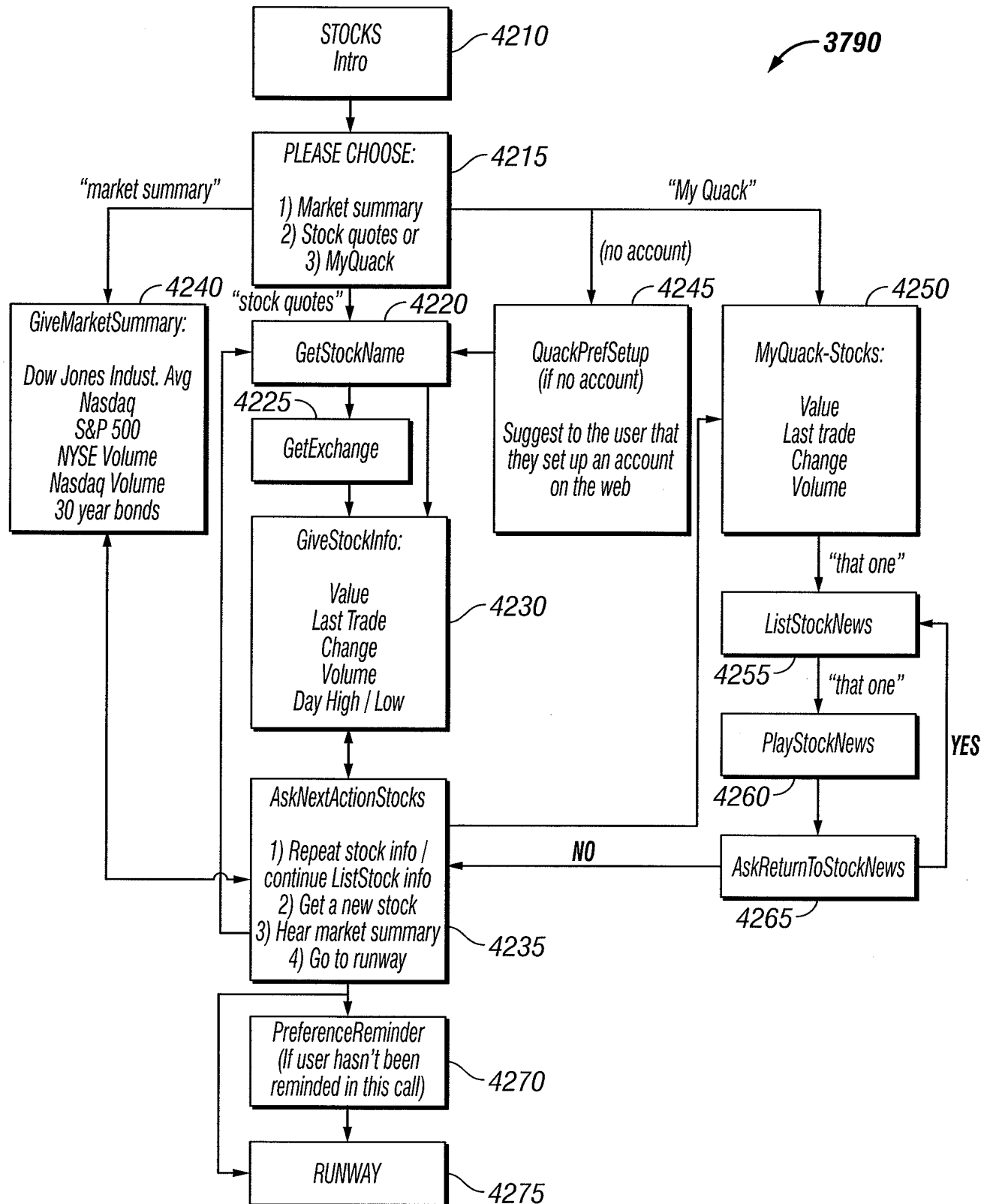
FIG. 37

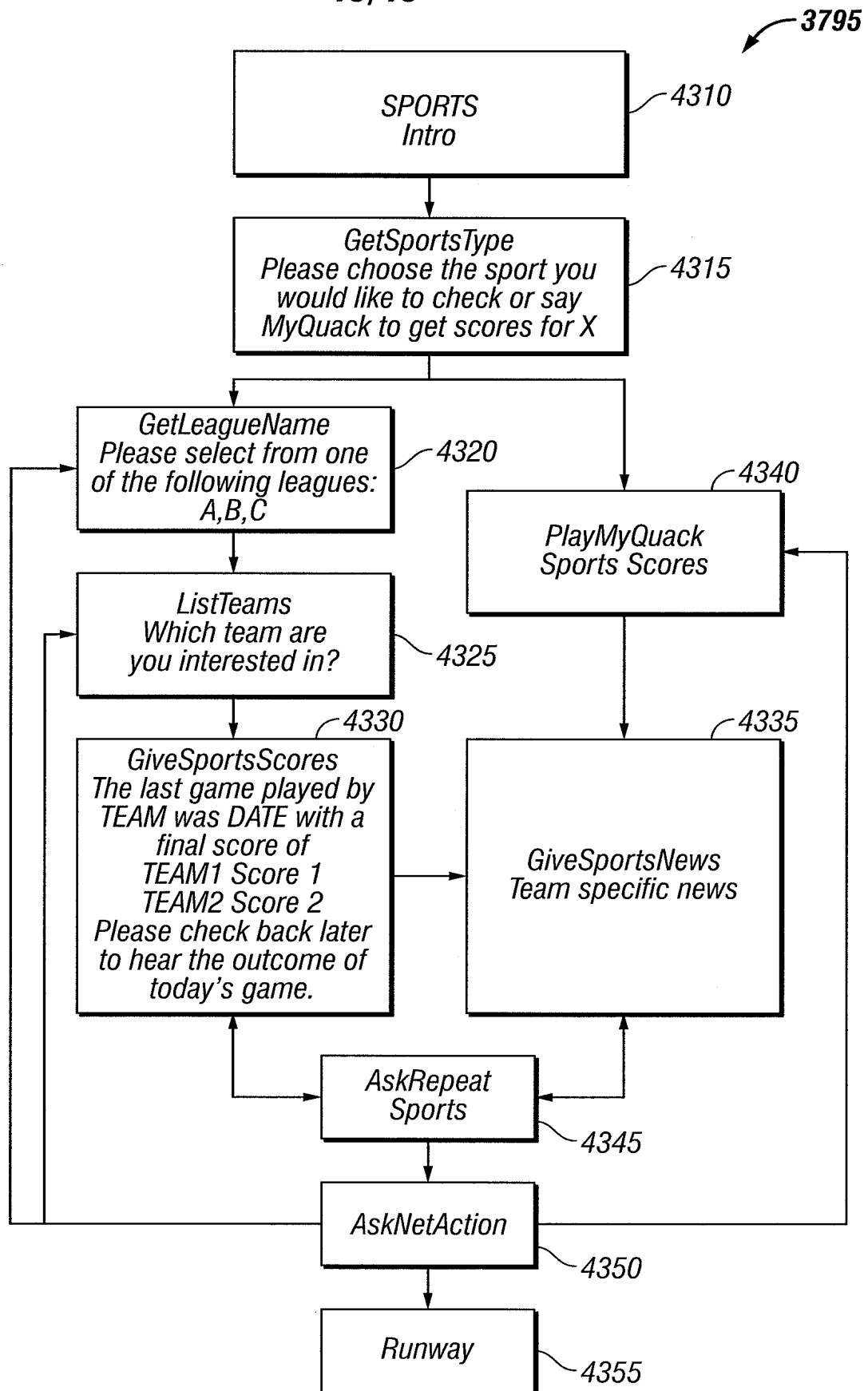
**35/40****FIG. 38**

**36/40**

**37/40****FIG. 40**

**38/40****FIG. 41**

**39/40****FIG. 42**

**40/40****FIG. 43**